NASA Technical Memorandum 78779

NASA-TM-78779

DONBOL: A Computer Program for Predicting Axisymmetric Nozzle Afterbody Pressure Distributions and Drag at Subsonic Speeds

Lawrence E. Putnam

MAY 1979

NASA

FEDD DOCUMENT

Note that this document bears the label "FEDD," an acronym for "FOR EARLY DOMESTIC DISSEMINATION." The FEDD label is affixed to documents that may contain information having high commercial potential.

The FEDD concept was developed as a result of the desire to maintain U.S. leadership in world trade markets and encourage a favorable balance of trade. Since the availability of tax-supported U.S. technology to foreign business interests could represent an unearned benefit, research results that may have high commercial potential are being distributed to U.S. industry in advance of general release.

The recipient of this report must treat the information it contains according to the conditions of the FEDD label on the front cover.

NASA Technical Memorandum 78779

DONBOL: A Computer Program for Predicting Axisymmetric Nozzle Afterbody Pressure Distributions and Drag at Subsonic Speeds

Lawrence E. Putnam
Langley Research Center
Hampton, Virginia

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Office

1979

SUMMARY

A Neumann solution for inviscid external flow has been coupled to a modified Reshotko-Tucker integral boundary-layer technique, the control volume method of Presz for calculating flow in the separated region, and an inviscid one-dimensional solution for the jet exhaust flow in order to predict axisymmetric nozzle afterbody pressure distributions and drag. The viscous and inviscid flows are solved iteratively until convergence is obtained. A computer algorithm of this procedure has been written and is called DONBOL. This paper provides a description of the computer program and a guide to its use. Comparisons of the predictions of this method with experiment show that the method accurately predicts the pressure distributions of boattail afterbodies which have the jet exhaust flow simulated by solid bodies. For nozzle configurations which have the jet exhaust simulated by high-pressure air, the present method significantly underpredicts the magnitude of nozzle pressure drag. This deficiency results because the method neglects the effects of jet plume entrainment. This method is limited to subsonic free-stream Mach numbers below that for which the flow over the body of revolution becomes sonic.

INTRODUCTION

The drag-producing components of the airplane propulsion system are usually installed in areas where the flow field is extremely complex. High body slopes and long boundary-layer runs, especially in the afterbody nozzle region, result in strong viscous effects on boattail drag. Furthermore, the viscous nature of the jet exhaust plume complicates the flow in this region. Because of these strong viscous interactions, current methods used for predicting the installed propulsion system drag are usually limited to empirical techniques. Recently, however, investigators have achieved some success in predicting uninstalled drag of axisymmetric nozzles with what is usually called the patched viscous-inviscid technique. (See refs. 1 to 7, for example.) In reference 1, Reubush and Putnam combine iteratively a conventional boundary-layer technique with a linearized potential-flow computation to account for the viscous-inviscid interaction. For boattail nozzles on which boundary-layer separation occurs, Reubush and Putnam employ the discriminating streamline concept of Presz (refs. 8 and 9) to separate the reverse flow region from the outer flow. The patched viscous-inviscid interaction methods have been successful in predicting the qualitative trends in boattail pressure drag with Mach number, Reynolds number, and nozzle geometry in spite of the complexity of the flow even for isolated boattails. (See ref. 1, for example.) In general, however, these techniques substantially underpredict the absolute levels of pressure drag on boattail nozzles at subsonic speeds.

Recently, an improved analytical model of the flow in the separated region has been developed by Presz (refs. 10 and 11). With this analytical model, the effects of axial-pressure gradients, surface skin friction, and jet plume entrainment on the shape of the discriminating streamline are computed. Predictions made using this new technique (refs. 10 and 11) are in substantially better agreement with experiment than the predictions of the previous methods (refs. 1 to 7). This improved model of the separated flow region, therefore, has been combined iteratively with the inviscid linearized potential-flow solution described in reference 1.

The present paper describes the various components of the resulting computer algorithm called DONBOL. Also, this paper illustrates the prediction capabilities of the method by comparison with experimental data. A user's guide to the computer program is presented. The computer program may be obtained from COSMIC, Suite 112, Barrow Hall, University of Georgia, Athens, GA 30602.

SYMBOLS

The symbols used in the computer printouts are given in a separate column.

A	SREF	maximum cross-sectional area of body of revolution
B		compressibility correction factor (see eq. (6))
C_D		boattail pressure drag coefficient, $\text{Drag}/q_\infty A$
	CDF	skin-friction drag coefficient
	CDP	pressure drag coefficient
	CDT	total drag coefficient
C_p	CP	static pressure coefficient
c_f	CF	local skin-friction coefficient
D	D	maximum diameter of body of revolution
d_b	DB	base diameter
H	H	boundary-layer shape factor, δ^*/θ
L		length of body of revolution
	L	reference length
l		length of nozzle or boattail
M	MO	Mach number
NPR		ratio of jet total pressure to free-stream static pressure, $p_{t,jet}/p_\infty$
N_{Re}		Reynolds number based on distance from nose of model to start of boattail

p		static pressure
p_t	PT	total pressure
q_∞		free-stream dynamic pressure
R		gas constant
	RC	body radius corrected for δ^* and discriminating streamline
r	R	radial coordinate of cylindrical coordinate system with origin at nose of body of revolution
	RDS	radius of the discriminating streamline
r^*		radius of stream tube for Mach number of 1
T_t	TT	total temperature
V_r	VR	ratio of radial velocity to free-stream velocity
	VT	ratio of local velocity to free-stream velocity
V_x	VX	ratio of axial velocity to free-stream velocity
x	X	axial coordinate of cylindrical coordinate system with origin at nose of body of revolution
Δx		axial distance downstream of start of boattail
β		$= \sqrt{1 - M_\infty^2}$
γ		ratio of specific heats
	DEL	boundary-layer thickness
δ^*	DEL*	boundary-layer displacement thickness
	ETA	local flow angle
θ	THETA	boundary-layer momentum thickness

Subscripts:

a	analogous configuration (see eqs. (1) to (6))
des	design conditions of nozzle
e	exit
exp	experiment

jet jet

p predicted

s separation

∞ free stream

DESCRIPTION OF METHOD

The present analytical method has been developed to calculate the flow over axisymmetric boattail bodies at subsonic speeds. It is assumed that the flow is composed of a viscous layer near the body, an inviscid external flow, and, if present, an inviscid jet exhaust flow. (See fig. 1.) The effect of the viscous layer is accounted for by modifying the body shape with an appropriate displacement thickness. In the framework of this representation, any boundary-layer separation on the boattail or nozzle surface is accounted for by modifying the afterbody geometry and plume boundary.

Inviscid External Flow Solution

The Neumann solution of reference 12 for incompressible flow over bodies of revolution was used to calculate the inviscid external flow. Since this is a solution for incompressible flow, the compressibility correction of reference 13 was used to correct for Mach number effects. The incompressible flow field considered is that for an "analogous" configuration obtained by means of the affine coordinate transformation given by the following equations:

$$x_a = \frac{x}{\beta} \tag{1}$$

$$r_a = r \tag{2}$$

where

$$\beta = \sqrt{1 - M_\infty^2} \tag{3}$$

The calculated flow velocities of the analogous configuration are then corrected using the following equations:

$$V_x = \frac{V_{x,a}}{B^2} \tag{4}$$

$$V_r = \frac{\beta V_{r,a}}{B^2} \tag{5}$$

where

$$B = \sqrt{1 - M_\infty{}^2(1 + V_{x,a})} \tag{6}$$

The pressure coefficients are obtained from the corrected velocities by using the compressible Bernoulli equation and the isentropic flow relations. Experience to date indicates that this compressibility correction provides better agreement with experimental results for flow over boattails than the classic Goethert compressibility correction.

Because the inviscid outer flow solution is based on incompressible flow theory with a compressibility correction, the present method is limited to free-stream Mach numbers for which the flow is subsonic everywhere.

Inviscid Jet Exhaust Flow

To calculate the inviscid boundary of the jet exhaust flow, a procedure based on one-dimensional isentropic flow theory has been developed and is used in the present computer program, DONBOL. The procedure for calculating the radius of the inviscid jet plume at any axial location downstream of the nozzle exit is as follows. Initially, a shape for the jet plume boundary is assumed. Next, the pressure distribution along this boundary is calculated. Then, a new value of the radius at each axial location is ascertained by calculating the cross-sectional area required to expand isentropically from the flow conditions at the nozzle exit to the pressure on the boundary at that location. This new boundary is used in the next iteration as the guess. The equations used to compute the inviscid jet plume boundary from the flow conditions at the nozzle exit and the pressure distribution along the boundary are as follows:

$$\left(\frac{p_t}{p}\right)_{des} = \left(1 + \frac{\gamma - 1}{2} M_{des}^2\right)^{\gamma/(\gamma-1)} \tag{7}$$

$$p_e = q_\infty C_{p,e} + p_\infty \tag{8}$$

If

$$\frac{p_{t,jet}}{p_e} > \left(\frac{p_t}{p}\right)_{des}$$

then

$$M_{jet} = M_{des} \tag{9}$$

If

$$\frac{p_{t,jet}}{p_e} \leqq \left(\frac{p_t}{p}\right)_{des}$$

then the static pressure across the exit is assumed equal to the external static pressure at the exit and

$$M_{jet} = \sqrt{\frac{2}{\gamma - 1}\left[\left(\frac{p_{t,jet}}{p_e}\right)^{(\gamma-1)/\gamma} - 1\right]} \tag{10}$$

Then

$$\frac{r^*}{r_e} = \sqrt{M_{jet}\left(\frac{\gamma + 1}{2}\right)^{(\gamma+1)/2(\gamma-1)}\left(1 + \frac{\gamma - 1}{2} M_{jet}^2\right)^{-(\gamma+1)/2(\gamma-1)}} \tag{11}$$

Now at any given x-location downstream of nozzle exit since C_p is a function of x,

$$p = q_\infty C_p + p_\infty \tag{12}$$

$$M = \sqrt{\frac{2}{\gamma - 1}\left[\left(\frac{p_{t,jet}}{p}\right)^{(\gamma-1)/\gamma} - 1\right]} \tag{13}$$

and

$$\frac{r^*}{r} = \sqrt{M\left(\frac{\gamma + 1}{2}\right)^{(\gamma+1)/2(\gamma-1)}\left(1 + \frac{\gamma - 1}{2} M^2\right)^{-(\gamma+1)/2(\gamma-1)}} \tag{14}$$

Then

$$\frac{r}{r_e} = \frac{r^*/r_e}{r^*/r} \tag{15}$$

Also

$$V_{x,jet} = M \sqrt{\frac{\gamma R T_{t,jet}}{1 + \frac{\gamma - 1}{2} M^2}} \tag{16}$$

This procedure has been used to calculate the shape of the inviscid jet plume for the exhaust flow from a convergent nozzle at two nozzle pressure ratios. The procedure is compared in figure 2 with the predictions of the method of Salas (ref. 14) modified to account for pressure variation along the jet boundary. The identical longitudinal pressure distribution along the boundary of the jet was assumed with each method. However, slightly different pressure distributions were assumed for each nozzle pressure ratio. At NPR = 2.90, the shapes of the jet exhaust plume boundary predicted by the two methods are in very good agreement. At NPR = 5.03, the one-dimensional method does not agree as well with the method of Salas. As will be shown later, however, the one-dimensional method does provide a reasonable estimate of the effects of NPR on nozzle drag.

Viscous Flow

The properties of the viscous boundary layer (both attached and separated) and the location of any separation on the nozzle boattail are calculated using the methods and computer algorithm developed by Presz, King, and Buteau and described in reference 10. Presz, King, and Buteau computed the turbulent boundary-layer displacement-thickness distribution along the body with the method described in reference 15. This method is a modified version of the Reshotko-Tucker integral boundary-layer solution (ref. 16). A comparison of the predictions of this technique with the experimental measurements of Winter, Rotta, and Smith (ref. 17) at $M_\infty = 0.6$ and a Reynolds number, based on body length, of 9.85×10^6 is presented in figure 3.

If boundary-layer separation occurs on the boattail, the boundary-layer equations become singular at the separation point. To overcome this difficulty, Presz uses the concept of a discriminating streamline to separate the reverse flow region from the outer boundary-layer flow. This method, described in reference 10, accounts for the effects of axial-pressure gradients, surface skin friction, and viscous mixing in the jet exhaust flow on the shape of this discriminating streamline. Note that the present method does not account for the effects of viscous mixing downstream of the reattachment point.

The use of Presz's model of the separated region requires that some method be available for predicting the location of separation. Several methods are available. They include Presz's control volume technique (ref. 8), Goldschmied's criterion (ref. 18), a modified Page criterion (ref. 19), and Stratford's criterion (ref. 20). A discussion of the accuracy of the various separation location criteria is given in reference 21 by Abeyounis. Any of these methods can be used in the current computer program.

Viscous-Inviscid Interaction

Since the boundary-layer displacement thickness, the discriminating streamline shape, and the inviscid jet boundary are functions of the pressure distribution along the body and the jet boundary, the final converged solution must be obtained by iteration between the inviscid outer flow solution, the inviscid jet plume solution, and the viscous boundary-layer solution. The iteration algorithm used in the present method is shown in figure 4 and is as follows:

(1) Calculate the inviscid pressure distribution on the body of revolution.

(2) Calculate the inviscid jet plume boundary.

(3) Calculate the boundary-layer displacement thickness.

(4) Calculate the location of boundary-layer separation on the boattail. The separation location is calculated using the criteria selected by the user and is based on the pressure distribution and, in some cases, boundary-layer characteristics of the flow over the body. For the first iteration, a separation location will always be predicted. Ideally, the separation location should move aft with increasing number of iterations, and the separation region should essentially disappear as the solution approaches convergence for nozzles and flow conditions where no boundary-layer separation would occur. Unfortunately, with the available separation criteria, this separation region does not always disappear. It is suggested that a solution first be attempted assuming attached flow for nozzles when there is a question about whether or not separation occurs. If the solution diverges, the user can then assume that the flow is not attached, and the calculation must be repeated assuming that the flow is separated.

(5) If a separated flow calculation is required, calculate the shape of the discriminating streamline. To speed convergence and to eliminate some initial numerical stability problems, the present method assumes that for the first four iterations, axial-pressure gradients do not affect the shape of the discriminating streamline. After nine iterations, the shape of the discriminating streamline is frozen.

will it converge if not frozen?

(6) Correct the body geometry for boundary-layer displacement effects by adding an effective displacement thickness to the original body. The effective displacement thickness includes the discriminating streamline in the separation region. A relaxation procedure described in reference 8 is used to expedite convergence and to eliminate instabilities in the iteration procedure.

(7) Repeat steps (1) to (6) for the desired number of iterations. In the present algorithm, no convergence criteria are specified. Convergence is assumed to occur when two successive iterations plotted to a reasonable scale give essentially the same results. To obtain this result, most configurations require about 15 iterations.

COMPARISONS OF PREDICTIONS AND EXPERIMENT

The predictions of program DONBOL for an $l/D = 1.768$, $d_b/D = 0.51$ circular-arc afterbody with a solid cylindrical jet plume simulator and with attached boundary-layer flow are compared with the experimental data of reference 22 in figure 5. At both free-stream Mach numbers shown, the agreement between the predicted and experimental pressure distributions is excellent. The boattail pressure drag of the configuration is underpredicted. However, the differences between theory and experimental drag are within the accuracy of the experimental measurements. These results are typical of all attached-flow cases computed to date with DONBOL.

factor of two different

For boattail nozzles and afterbodies on which the boundary layer separates, the agreement between the predictions of the present method and experiment depends on the chosen separation criterion. In reference 21 Abeyounis showed that a criterion predicts significantly different locations for separation depending on whether the theoretical inviscid pressure distribution or the experimental pressure distribution is used. This result suggests that the predicted separation location may also be a function of the iteration algorithm used in a patched viscous-inviscid interaction procedure such as DONBOL. Therefore, the accuracy of a given separation criterion should be assessed using the total prediction algorithm for which it is to be incorporated. Predictions of the separation location criteria incorporated in DONBOL are compared with the experimental data of Abeyounis (ref. 21) in figure 6(a). The large differences shown in predicted separation location can affect predicted afterbody pressure distributions and drag significantly, as illustrated in figure 6(b). Based on these limited results and because it more accurately predicts the location of separation on the steep, highly separated $l/D = 0.8$ boattail configuration, the method of Presz is recommended and is used for all further calculations presented in this paper.

An illustration of the capabilities of the present method for predicting the effects of free-stream Mach number on the pressure distribution and drag of an afterbody with separated boundary layer is shown in figure 7. The experimental data from references 22 and 23 shown in this figure are for the same $l/D = 0.8$, $d_b/D = 0.51$ circular-arc afterbody with solid cylindrical plume simulator for which separation location data are presented in figure 6(a). At a Mach number of 0.4 where the separation location is accurately predicted using the Presz criterion, the agreement between experimental and predicted pressure distributions is very good. As the difference in predicted and actual separation location increases with increasing free-stream Mach number, the agreement in pressure distributions between theory and experiment deviates somewhat. However, as shown in figure 7(b) the agreement between predicted and actual boattail pressure drag improves with increasing Mach number. This agreement is essentially within experimental accuracy throughout the range of Mach numbers for which the theory is applicable.

as assumption gets worse, results get better?

says it doesn't matter anyhow

At a given free-stream Mach number, the agreement between theory and experiment is a function of boattail geometry. The comparisons between the theory and experiment of reference 23 shown in figure 8 indicate that for boattails with less closure than the configuration of figure 7, substantially better agreement between theory and experiment can result.

The capabilities of the present method for predicting the effects of Reynolds number on boattail pressure distributions and drag are illustrated in figure 9. The agreement between theory and experiment is a function of Reynolds number and boattail geometry. However, the predicted variation of the boattail-pressure drag coefficient with Reynolds number is in relatively good agreement with the experimental results (fig. 9(c)).

A comparison of the experimental (refs. 22 to 24) and predicted effects of the ratio of nozzle total pressure to free-stream static pressure NPR on the pressure distribution and drag of a $l/D = 0.8$, $d_b/D = 0.51$ circular-arc nozzle is shown in figure 10. In general, the present method reasonably predicts the variation of the pressure distributions with NPR. However, at both $M_\infty = 0.6$ and 0.8 (figs. 10(a) and 10(b)) DONBOL generally predicts more positive pressures than actually exist on the nozzle. As a result, the magnitude of the boattail drag is substantially underpredicted (fig. 10(c)). These deficiencies probably result because the present method does not account for the effects of jet entrainment. Note that the present method does account for the effects of jet entrainment on the shape of the separation discriminating streamline, but does not account for jet plume entrainment in any manner downstream of the reattachment of the separated boundary layer. Jet entrainment downstream of reattachment should reduce the pressures on the nozzle and thereby increase the nozzle drag. As shown in figure 10(c), the present method accurately predicts the nozzle drag when the jet exhaust flow is simulated experimentally by a solid cylindrical sting. This solid sting, of course, does not simulate the effects of jet plume entrainment, but does simulate the effects of jet plume blockage on the flow over the nozzle. Even though the present method does not accurately predict the magnitude of nozzle drag, it does predict the decrease in drag at the higher nozzle total-pressure ratios. The present method does not, however, predict the increase in drag at the lower pressure ratios. Further illustrations of the capabilities of the present method for predicting pressure distributions and drag for nozzles with jet exhaust flow are shown in figure 11. Here the program DONBOL was used to calculate the flow over the equivalent bodies of reference 25 with the nozzles operating at an NPR of approximately 2.5. For these configurations, the predictions generally agree better with experiment than for the configuration shown in figure 10.

DESCRIPTION OF COMPUTER PROGRAM

A flow chart of program DONBOL is presented in figure 12 and a listing of the program is provided in the appendix. This program is written in overlay form and consists of the main overlay and four primary overlays. Primary overlays 1 to 3 are used to calculate the inviscid external flow, and overlay (5,0) is used to calculate the inviscid jet exhaust flow, the boundary-layer flow, and the "effective" body geometry for further iterations. The program uses nine disk files during computation. Input data are obtained from TAPE5 and the results are written on TAPE6 which is set equal to OUTPUT. A restart output file is written on TAPE7. The remaining disk files are used internally by the program. DONBOL requires about 125 000 octal storage locations on the Control Data CYBER 175 computer system and executes 15 iterations in approximately 3 minutes.

A brief description of the various routines in the program is given in the following list:

DONBOL This routine reads the input data, stores the x- and r-coordinates on TAPE13, and controls the iteration procedure. All primary overlays are called from this routine.

ONE This routine prints certain control parameter information and calls subroutines BASIC1 and MATRIX.

BASIC1 This subroutine makes the compressibility correction transformations to the x- and r-coordinates, calculates coordinates of the midpoint of each body panel, and calculates the slope of each body panel.

MATRIX The influence coefficient matrix and the boundary condition matrix are set up in this subroutine. MATRIX calls subroutine XYZ.

XYZ The influence coefficients are calculatd by this subroutine. A constant source of unit strength is assumed to act on each panel. The influence coefficient is the integral of the effect of the constant strength source. The subroutine calls XYZ1 and XYZ2.

XYZ1 This subroutine performs the integration of the effects of the constant strength source for points within a specified radius of the singularity.

XYZ2 This subroutine performs the integration using Simpson's rule to determine the influence of the unit source panel at all distances greater than the specified radius from the singularity. The routine calls subroutine ELIP.

ELIP This subroutine is used to calculate the value of various elliptical integrals.

TWO This routine initializes parameters for call to MISNA2.

MISNA2 This subroutine calculates the strengths of the source panels by solving the matrix equation using a Seidel iteration procedure.

THREE This routine initializes various parameters and then calls subroutine AXIS. The pressure coefficients computed by AXIS are then written on TAPE13.

AXIS This subroutine calculates velocity components of the flow and surface-pressure coefficients. The velocity components are corrected for compressibility effects using either the Goethert or Labrujere method, before computing the pressure coefficient.

FIVE This routine is the interface between the inviscid external flow calculation and the viscous flow calculations. The body geometry and pressure coefficients are read from TAPE13. The inviscid jet plume exhaust flow boundary and velocity are calculated. The viscous subroutine package is called to obtain boundary-layer parameters and the corrected effective body contour. See reference 8 for details of the subroutines in the viscous package. Routine FIVE also computes the drag coefficient. The results are printed and the final solution put on TAPE7 for further iteration if necessary.

IUNI This is a Langley Research Center computer system library subroutine. The subroutine uses first- or second-order Lagrangian interpolation to estimate the value of a set of functions at a specified value of the independent value.

Description of Input Data Cards

Sample input data required for program DONBOL are presented in figure 13. This figure presents the input data required to compute the flow over a boattail nozzle configuration with jet exhaust flow. This test case also illustrates the input data required to compute flow conditions at points off the body. Specifically, the input data required are as follows:

Card 1: identification.- Card 1 contains any desired identifying information in columns 1 to 80.

Card 2: control integers.- Card 2 contains 13 integers, each punched right justified in a five-column field. An identification of the card columns, the name used by the source program, and a description of each integer is given in the following table:

Columns	FORTRAN name	Description
1 to 5	ISWITCH	Calculation Option Code: If ISWITCH = 1, potential-flow solution only. If ISWITCH = 2, boundary-layer effects on pressure distribution are included in solution using an iteration scheme. If ISWITCH = 3, boundary-layer solution only.
6 to 10	IPRINT	Iteration number to start printing results.
11 to 15	IPUNCH	Punch option code: If IPUNCH greater than 0, last iteration is written on TAPE7 in format necessary for a restart of solution. CP for last iteration also written on TAPE7.
16 to 20	ITERA	Iteration number for first calculation of this submittal. For initial submittal of any calculation, ITERA must be 0.

Columns	FORTRAN name	Description
21 to 25	ITERMAX	Maximum number of iterations (less than or equal to 20).
26 to 30	IMACH	Compressibility correction code: If IMACH = 1, Goethert compressibility correction used. If IMACH = 2, Labrujere compressibility correction used.
31 to 35	ISEP	Separation location criteria code: If ISEP = 0, no separation model used. If ISEP = 1, separation location specified by user. ! If ISEP = 2, Presz control volume criterion used. If ISEP = 3, Goldschmied criterion used. If ISEP = 4, modified Page criterion used. If ISEP = 5, Stratford criterion used.
41 to 45	INT(3)	X-array location to start search for separation.
46 to 50	INT(4)	X-array location to end search for separation.
51 to 55	INT(5)	Jet plume and entrainment option: If INT(5) = 0, omit jet plume and entrainment calculations. If INT(5) = 1, include jet plume and entrainment calculations.
56 to 60	INT(6)	X-array location of nozzle exit.
61 to 65	INT(7)	Smoothing parameter: If INT(7) = 0, no smoothing. If INT(7) = 1, aerodynamic body contour and pressure distribution are smoothed. INT(7) = 1 should be used. !
66 to 70	IFLAG5	An integer which if greater than 0 specifies that off-body points are to be calculated.

<u>Card 3: free-stream conditions and reference dimensions</u>.- Card 3 contains quantities used to define the free-stream flow and dimensional information required to convert body coordinate inputs to meters. If the separation location is to be input by the user, it is given on this card. Identification of the card columns, names used in the source program, and a description of each variable is given in the following table:

Columns	FORTRAN name	Description
1 to 10	MO	Free-stream Mach number.
11 to 20	PT	Free-stream total pressure, Pa.
21 to 30	TT	Free-stream total temperature, K.
31 to 40	REFL	Reference length - factor required to convert input values of x and r to meters.
41 to 50	SREF	Reference area, meters^2.
51 to 60	XSEPND	The x-coordinate of the separation location. Required if ISEP = 1.

Card 4: jet exhaust conditions.- This card contains quantities used to define the jet exhaust flow. If there is no jet exhaust flow (INT(5) = 0) this card may be blank, but it must be input. The card contains the following information:

Columns	FORTRAN name	Description
1 to 10	XMJET	Mach number of jet at nozzle exit.
11 to 20	PTJET	Jet total pressure, Pa.
21 to 30	TTJET	Jet total temperature, K.
31 to 40	RJET	Radius of nozzle exit.

Cards 5, 6, . . .: remaining data input cards.- The remaining data cards provide a description of the body geometry, the location of any off-body points at which the flow is to be calculated, and the surface pressure coefficients if the boundary-layer solution only is to be computed. Unless otherwise noted, each card contains up to six values with each value punched in a ten-column field with a decimal.

Body geometry cards: The first body geometry data card gives the number of coordinates, NN. The integer, NN is punched in columns 1 to 5 right justified. The number of body coordinates may not be greater than 200. The next group of body geometry data cards contains the axial location at which the body radius is to be specified. There are exactly NN locations with up to six values per card. The next group of body geometry data cards contains the radius of the body at the specified axial locations. Again there are NN values of the body radius specified. Note that if the jet exhaust flow option is selected, an initial guess of the shape of the jet plume boundary must be included in the description of the body geometry.

Off-body points: If the flow is to be calculated at any off-body points *and IFLAG5 > 0, then the following cards must be input. First the number of* off-body points must be specified on a data card. The number of off-body points is punched in columns 1 to 5 right justified. (Note that the sum of the points on the body of revolution and the off-body points may not be greater than 200.) Then a group of data cards giving the location of the x-coordinates at which the flow is to be calculated is input. This group of cards is followed by a group of cards on which the r-coordinates of the off-body points are specified.

Pressure coefficients cards: This group of cards is input only if the program is to be restarted or if ISWITCH = 3, that is, when the boundary-layer solution only is to be calculated. The pressure coefficient at each body x-coordinate location is input with six values per card.

Description of Output

Program output consists of printed output and a disk file TAPE7 written in the form necessary for a restart of the program. An example of the printed output is presented in figure 14 for the test case presented in figure 13.

The first page of output includes the program title, case identification, list of control options selected, free-stream conditions, and, if requested, jet exhaust flow conditions. On the second page, several diagnostic messages from various routines in the program are written.

Following these pages, the results of the calculation are output. Case identification and free-stream conditions are again specified. The iteration number, the reference length L, the reference area SREF, and the axial location of boundary-layer separation and reattachment are given. Following this information, tabulated listings of the body axial coordinate X/L, the body radial coordinate R/L, the body radius corrected for the discriminating streamline RDS/L, and the body radius corrected for boundary-layer displacement thickness and the discriminating streamline RC/L are printed. Also listed are values of pressure coefficient CP, local skin-friction coefficient CF, boundary-layer thickness DEL/L, boundary-layer displacement thickness DEL*X/L, boundary-layer momentum thickness THETA/L, and boundary-layer shape factor H. In addition, listings of the pressure drag coefficient CDP, skin-friction drag coefficient CDF, and total drag coefficient CDT are given. The drag values listed are based on the reference area SREF and are the integrals of the pressure forces and/or skin-friction forces from the nose of the body to the specified X/L location. To obtain the nozzle boattail pressure drag coefficient, for example, it is necessary to subtract the value of the pressure drag coefficient at the start of the boattail from the value of the pressure drag coefficient at the nozzle exit or end of the boattail. This information is repeated for each iteration as specified in the input data.

If flow conditions at off-body points are calculated, the axial location X/L and radial location R/L of the off-body points are tabulated on the next page together with the ratio of axial velocity to free-stream velocity VX, the ratio of radial velocity to free-stream velocity VR, and the ratio of local velocity to free-stream velocity VT. Also tabulated are the local flow angle ETA in radians, the local Mach number ML, and the local pressure coefficient CP.

CONCLUDING REMARKS

A computer program has been written to compute the flow over axisymmetric nozzle configurations at subsonic speeds with and without separated flow. The computer algorithm is based on a patched viscous-inviscid interaction procedure. That is, solutions for the various regions of the flow are coupled together and solved iteratively to obtain a converged solution. The results of the present alogrithm called DONBOL are in good agreement with experimental pressure distribution results for flow over nozzles with the jet exhaust simulated with solid bodies. The method substantially underpredicts the magnitude of the boattail drag when the jet exhaust flow is simulated with high-pressure air. This deficiency results because the present technique does not account for the effects of jet plume entrainment downstream of reattachment of the separated boundary layer on the flow over the nozzle. The method is limited to free-stream Mach numbers below that for which flow on the body of revolution reaches sonic speeds.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
April 11, 1979

APPENDIX

TABULATED LISTING OF COMPUTER PROGRAM

APPENDIX

```
      OVERLAY(LINK,0,0)
      PROGRAM DONBOL(INPUT=201,OUTPUT=201,TAPE5,TAPE6=OUTPUT,TAPE3=1001,DON
     1TAPE4=1001,TAPE9=1001,TAPE11=1001,TAPE12=1001,TAPE13=1001,TAPE7=10DON
     201)                                                               DON
C                                                                       DON
C                                                                       DON
C     ******************************************************************DON
C                                                                      *DON
C             INPUT DATA DESCRIPTION FOR PROGRAM DONBOL                *DON
C             -------------------------------------------              *DON
C                                                                      *DON
C     CARD    COL        NAME                  DESCRIPTION             *DON
C                                                                      *DON
C       1     1-80       HEDR       CASE DESCRIPTION                   *DON
C                                                                      *DON
C       2     1-5       ISWITCH     CALCULATION OPTION CODE -          *DON
C                                   IF ISWITCH=1 POTENTIAL FLOW SOLUTION *DON
C                                     ONLY.                            *DON
C                                   IF ISWITCH=2 BOUNDARY LAYER EFFECTS ON*DON
C                                     PRESSURE DISTRIBUTION ARE INCLUDED *DON
C                                     IN SOLUTION USING AN ITERATION   *DON
C                                     SCHEME.                          *DON
C                                   IF ISWITCH=3 BOUNDARY LAYER SOLUTION *DON
C                                     ONLY.                            *DON
C                                                                      *DON
C       2     6-10       IPRINT     ITERATION NUMBER TO START PRINTING *DON
C                                   RESULTS.                           *DON
C                                                                      *DON
C       2    11-15       IPUNCH     PUNCH OPTION CODE - IF IPUNCH GREATER *DON
C                                   THAN 0 LAST ITERATION ON TAPE 7 IN *DON
C                                   FORMAT NECESSARY FOR A RESTART OF  *DON
C                                   SOLUTION.  CP FOR LAST ITERATION ALSO *DON
C                                   WRITTEN ON TAPE 7.                 *DON
C                                                                      *DON
C       2    16-20       ITERA      ITERATION NUMBER FOR FIRST CALCULATION*DON
C                                   OF THIS SUBMITTAL                  *DON
C                                                                      *DON
C       2    21-25      ITERMAX     MAXIMUM NUMBER OF ITERATIONS (LESS *DON
C                                   THAN OR EQUAL TO 20)               *DON
C                                                                      *DON
C                                                                      *DON
C                                                                      *DON
C     CARD    COL        NAME                  DESCRIPTION             *DON
C                                                                      *DON
C       2    26-30       IMACH      COMPRESSIBILITY CORRECTION CODE -  *DON
C                                   IF IMACH=1 GOETHERT COMPRESSIBILITY *DON
C                                     CORRECTION USED.                 *DON
C                                   IF IMACH=2 LABRUJERE COMPRESSIBILITY *DON
C                                     CORRECTION USED.                 *DON
C                                                                      *DON
C       2    31-35        ISEP      SEPARATION LOCATION CRITERIA CODE - *DON
C                                   IF ISEP=0 NO SEPARATION MODEL USED. *DON
C                                   IF ISEP=1 SEPARATION LOCATION      *DON
C                                     SPECIFIED BY USER.               *DON
C                                   IF ISEP=2 PRESZ CONTROL VOLUME     *DON
C                                     CRITERIA USED.                   *DON
C                                   IF ISEP=3 GOLDSCHMIED CRITERIA USED. *DON
C                                   IF ISEP=4 MODIFIED PAGE CRITERIA USED.*DON
C                                   IF ISEP=5 STRATFORD CRITERIA USED. *DON
C                                                                      *DON
C       2    36-40                  NOT USED.                          *DON
C                                                                      *DON
C       2    41-45       INT(3)     X-ARRAY LOCATION TO START SEARCH FOR *DON
C                                   SEPARATION.                        *DON
C                                                                      *DON
C       2    46-50       INT(4)     X-ARRAY LOCATION TO END SEARCH FOR *DON
```

```
C                                   SEPARATION.                             *DON  66
C                                                                           *DON  67
C      2   51-55     INT(5)         JET PLUME AND ENTRAINMENT OPTION -      *DON  68
C                                   IF INT(5)=0 OMIT JET PLUME AND          *DON  69
C                                     ENTRAINMENT CALCULATIONS.             *DON  70
C                                   IF INT(5)=1 INCLUDE JET PLUME AND       *DON  71
C                                     ENTRAINMENT CALCULATIONS.             *DON  72
C                                                                           *DON  73
C      2   56-60     INT(6)         X-ARRAY LOCATION OF NOZZLE EXIT.        *DON  74
C                                                                           *DON  75
C      2   61-65     INT(7)         SMOOTHING PARAMETER -                   *DON  76
C                                   IF INT(7)=0 NO SMOOTHING.               *DON  77
C                                   IF INT(7)=1 AERODYNAMIC BODY CONTOUR    *DON  78
C                                   AND PRESSURE DISTRIBUTION ARE SMOOTHED*DON  79
C                                                                           *DON  80
C      2   66-70     IFLAGS         AN INTEGER WHICH IF GREATER THAN 0      *DON  81
C                                   SPECIFIES THAT OFF BODY POINTS ARE TO   *DON  82
C                                   BE CALCULATED.                          *DON  83
C                                                                           *DON  84
C    CARD   COL       NAME                    DESCRIPTION                   *DON  85
C                                                                           *DON  86
C      3    1-10       MN           FREE STREAM MACH NUMBER.                *DON  87
C                                                                           *DON  88
C      3   11-20       PT           FREE STREAM TOTAL PRESSURE, PASCALS     *DON  89
C                                                                           *DON  90
C                                                                           *DON  92
C      3   21-30       TT           FREE STREAM TOTAL TEMPERATURE, KELVIN  *DON  91
C      3   31-40      REFL          REFERENCE LENGTH - FACTOR REQUIRED TO  *DON  93
C                                   CONVERT INPUT VALUES OF X AND R TO      *DON  94
C                                   METERS.                                 *DON  94A
C                                                                           *DON  95
C      3   41-50      SREF          REFERENCE AREA, SQ METERS               *DON  96
C                                                                           *DON  97
C      3   51-60     XSEPND         THE X-COORDINATE OF THE SEPARATION      *DON  98
C                                   LOCATION.  REQUIRED IF ISEP=1.          *DON  99
C                                                                           *DON 100
C      4    1-10     XMJET          MACH NUMBER OF JET AT NOZZLE EXIT.      *DON 101
C                                                                           *DON 102
C      4   11-20     PTJET          JET TOTAL PRESSURE, PASCALS             *DON 103
C                                                                           *DON 104
C      4   21-30     TTJET          JET TOTAL TEMPERATURE, KELVIN           *DON 105
C                                                                           *DON 106
C      4   31-40      RJET          RADIUS OF NOZZLE EXIT                   *DON 107
C                                                                           *DON 108
C      5    1-5        NN           NUMBER OF COORDINATES FOR BODY          *DON 109
C                                                                           *DON 110
C      6    1-60   X(I),I=1,NN      THE X-COORDINATES OF THE POINTS DEFIN-*DON 111
C                                   ING THE BODY.  DATA IS INPUT WITH A     *DON 112
C                                   FORMAT OF 6F10.6.  MAY BE MORE THAN     *DON 113
C                                   ONE CARD                                *DON 114
C                                                                           *DON 115
C      7    1-60   R(I),I=1,NN      THE R-COORDINATES OF THE POINTS DEFIN-*DON 116
C                                   ING THE BODY.  DATA IS INPUT WITH A     *DON 117
C                                   FORMAT OF 6F10.6.  MAY BE MORE THAN     *DON 118
C                                   ONE CARD                                *DON 119
C                                                                           *DON 120
C                                                                           *DON 121
C                                                                           *DON 122
C                                                                           *DON 123
C    IF IFLAGS GREATER THAN 0 THE FOLLOWING CARDS MUST BE INPUT             *DON 124
C                                                                           *DON 125
C    CARD   COL       NAME                    DESCRIPTION                   *DON 126
C                                                                           *DON 127
C      8    1-5        NN           NUMBER OF OFF BODY POINTS               *DON 128
C                                                                           *DON 129
C      9    1-60   X(I),I=1,NN      THE X-COORDINATES OF THE OFF BODY       *DON 130
```

```
C                                POINTS.  DATA IS INPUT WITH A FORMAT  *DON 131
C                                OF 6F10.6.  MAY BE MORE THAN ONE CARD.*DON 132
C                                                                      *DON 133
C     10    1-60  R(I),I=1,NN    THE R-COORDINATES OF THE OFF BODY     *DON 134
C                                POINTS.  DATA IS INPUT WITH A FORMAT  *DON 135
C                                OF 6F10.6.  MAY BE MORE THAN ONE CARD.*DON 136
C                                                                      *DON 137
C     IF ISWITCH IS EQUAL TO 3 THE FOLLOWING CARD MUST BE INPUT        *DON 138
C                                                                      *DON 139
C     CARD   COL       NAME                  DESCRIPTION               *DON 140
C                                                                      *DON 141
C     11    1-60  CP(I),I=1,NN   PRESSURE COEFFICIENT AT EACH X-       *DON 142
C                                COORDINATE ON BODY.  DATA IS INPUT    *DON 143
C                                WITH A FORMAT OF 6F10.6.  MAY BE MORE *DON 144
C                                THAN ONE CARD.                        *DON 145
C                                                                      *DON 146
C     *****************************************************************DON 147
C                                                                       DON 148
C                                                                       DON 149
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLADON 150
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,DON 151
     2AMJET,PTJET,TTJET,RJET,RSTAR                                      DON 152
      COMMON /SAVE/ VDUM(951)                                           DON 153
      DIMENSION X(200), R(200), CP(200)                                 DON 154
      INTEGER FLG05,BDN                                                 DON 155
      REAL MN                                                           DON 156
C                                                                       DON 157
      LINK=4LLINK                                                       DON 158
      BDN=1                                                             DON 159
      DO 10 I=1,200                                                     DON 160
10    CP(I)=0.0                                                         DON 161
20    READ (5,70) HEDR                                                  DON 162
      IF (EOF(5)) 60,30                                                 DON 163
30    READ (5,80) ISWITCH,IPRINT,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,(INT(I)DON 164
     1,I=2,7),IFLAG5                                                    DON 165
      READ (5,90) MN,APT,ATT,REFL,SREF,XSEPND,AMJET,PTJET,TTJET,RJET    DON 166
      READ (5,80) NN                                                    DON 167
      READ (5,90) (X(I),I=1,NN)                                         DON 168
      DO 40 IZ=1,2                                                      DON 169
      IF (IZ.EQ.1.OR.ITERA.GT.0) READ (5,90) (R(I),I=1,NN)              DON 170
      WRITE (13) NN,(X(I),I=1,200)                                      DON 171
      WRITE (13) NN,(R(I),I=1,200)                                      DON 172
40    CONTINUE                                                          DON 173
      IF ((ISWITCH.EQ.3).OR.(ITERA.GT.0)) READ (5,90) (CP(I),I=1,NN)    DON 174
      WRITE (13) (CP(I),I=1,200)                                        DON 175
      IF (ISWITCH.EQ.3) CALL OVERLAY (LINK,5,0)                         DON 176
      IF ((ISWITCH.EQ.1).OR.(ISWITCH.EQ.2)) GO TO 50                    DON 177
      GO TO 20                                                          DON 178
50    IF (ITERA.GT.ITERMAX.AND.IFLAG5.EQ.0) GO TO 20                    DON 179
      REWIND 3                                                          DON 180
      REWIND 4                                                          DON 181
      REWIND 9                                                          DON 182
      REWIND 11                                                         DON 183
      REWIND 12                                                         DON 184
      REWIND 13                                                         DON 185
      CALL OVERLAY (LINK,1,0)                                           DON 186
      CALL OVERLAY (LINK,2,0)                                           DON 187
      CALL OVERLAY (LINK,3,0)                                           DON 188
      IF (IFLAG5.GT.0.AND.ITERA.GE.(ITERMAX+1)) GO TO 20                DON 189
      CALL OVERLAY (LINK,5,0)                                           DON 190
      ITERA=ITERA+1                                                     DON 191
      GO TO 50                                                          DON 192
60    CONTINUE                                                          DON 193
      STOP                                                              DON 194
C                                                                       DON 195
C                                                                       DON 196
```

```
70    FORMAT (8A10)
80    FORMAT (16I5)
90    FORMAT (6F10.6)
      END
      OVERLAY(LINK,1,0)
      PROGRAM ONE
C
C              *CONTROL FOR BASIC DATA AND FORM MATRIX
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ TX1(200),TY1(200),NG(200),TG(200),ALFA(200),RSDS(200),
     1DALF(200),TEMP(1017)
      INTEGER FLG05,BDN
      REAL MN,NG
C
C         OUTPUT CASE CONTROL DATA
C
      FLG05=0
      IF (ITERA.GT.0) GO TO 10
      WRITE (6,250)
      WRITE (6,20) HEDR
      IF (IFLAG5.GT.0) WRITE (6,30)
      IF (IMACH.EQ.1) WRITE (6,40)
      IF (IMACH.EQ.2) WRITE (6,50)
      WRITE (6,60)
      IF (ISEP.EQ.0) WRITE (6,70)
      IF (ISEP.GT.0) WRITE (6,80)
      IF (ISEP.EQ.1) WRITE (6,90) XSEPND
      IF (ISEP.EQ.2) WRITE (6,100)
      IF (ISEP.EQ.3) WRITE (6,110)
      IF (ISEP.EQ.4) WRITE (6,120)
      IF (ISEP.EQ.5) WRITE (6,130)
      IF (ISEP.GE.2) WRITE (6,140) INT(3)
      IF (ISEP.GE.2) WRITE (6,150) INT(4)
      IF (INT(5).GT.0) WRITE (6,160)
      WRITE (6,170) INT(6)
      IF (INT(7).GT.0) WRITE (6,180)
      WRITE (6,190)
      G=1.4
      G1=(G-1.)/2.
      G2=G/(G-1)
      RG=286.96
      PO=APT/(1.+G1*MN**2)**G2
      TO=ATT/(1.+G1*MN**2)
      RHO=PO/(TO*RG)
      XMU=1.458/10**6*TO**1.5/(TO+110.33)
      RN=RHO*MN*SQRT(G*RG*TO)/XMU
      RN=RN/10.**6
      WRITE (6,200)
      WRITE (6,210) MN,APT,ATT
      WRITE (6,220) RN
      IF (INT(5).GT.0) WRITE (6,240)
      IF (INT(5).GT.0) WRITE (6,210) AMJET,PTJET,TTJET
      XNPR=PTJET/PO
      IF (INT(5).GT.0) WRITE (6,230) XNPR
      IF (IPRINT.GT.0) WRITE (6,250)
C
C         SETUP FOR UNIFORM FLOW
C
10    CALL BASIC1
      NSIGA=1
      REWIND 4
```

```
      CALL MATRIX
C
C
C
20    FORMAT (10X,64HDONBOL --- AN AXISYMMETRIC INVISCID/VISCID INTERA
     1CTION PROGRAM//16X,52HBY LAWRENCE E. PUTNAM, NASA, LANGLEY RESEARC
     2H CENTER//2X,13HCASE TITLE = ,8A10//13X,29H***** CASE CONTROL DATA
     3 *****//)
30    FORMAT (13X,15HOFF-BODY POINTS)
40    FORMAT (13X,35HGOETHERT COMPRESSIBILITY CORRECTION)
50    FORMAT (13X,36HLABRUJERE COMPRESSIBILITY CORRECTION)
60    FORMAT (13X,48HMODIFIED RESHOTKO TUCKER BOUNDARY LAYER SOLUTION)
70    FORMAT (13X,29HSEPARATED FLOW MODEL NOT USED)
80    FORMAT (13X,64HPRESZ MODIFIED CONTROL VOLUME DISCRIMINATING STREAM
     1LINE SOLUTION)
90    FORMAT (13X,46HSEPARATION LOCATION SPECIFIED BY USER AT X/L =,F10.
     16)
100   FORMAT (13X,49HPRESZ CONTROL VOLUME SEPARATION LOCATION CRITERIA)
110   FORMAT (13X,40HGOLDSCHMIED SEPARATION LOCATION CRITERIA)
120   FORMAT (13X,42HMODIFIED PAGE SEPARATION LOCATION CRITERIA)
130   FORMAT (13X,38HSTRATFORD SEPARATION LOCATION CRITERIA)
140   FORMAT (13X,34HSTART SEARCH FOR SEPARATION AT I =,I4)
150   FORMAT (13X,32HEND SEARCH FOR SEPARATION AT I =,I4)
160   FORMAT (13X,29HJET EXHAUST PLUME CALCULATION)
170   FORMAT (13X,18HNOZZLE EXIT AT I =,I4)
180   FORMAT (13X,26HSMOOTH AERODYNAMIC CONTOUR)
190   FORMAT (13X,28HSMOOTH PRESSURE DISTRIBUTION)
200   FORMAT (1H0,12X,22HFREE STREAM CONDITIONS)
210   FORMAT (20X,20HMACH NUMBER         =,F12.3/20X,20HTOTAL PRESSURE
     1  =,F12.3,8H PASCALS/20X,20HTOTAL TEMPERATURE  =,F12.3,7H KELVIN)
220   FORMAT (20X,20HREYNOLDS NUMBER     =,F12.3,18H MILLION PER METER)
230   FORMAT (20X,20HNPR                 =,F12.3)
240   FORMAT (1H0,12X,37HJET EXHAUST CONDITIONS AT NOZZLE EXIT)
250   FORMAT (1H1)
      END
      SUBROUTINE BASIC1
C
C           * READ DATA AND SETUP FOR UNIFORM FLOW
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ TX1(200),TY1(200),NG(200),TG(200),ALFA(200),RSDS(200),
     1DALF(200),TEMP(1017)
      INTEGER FLG05,BDN
      REAL MN,NG
C
      REWIND 13
      NT=0
      K=0
      K2=1
      IF (ITERA.GT.ITERMAX) FLG05=IFLAG5
      IF (FLG05.NE.0) K2=2
C
C           * MAJOR LOOP * NO. OF BODIES + OFF BODY POINTS
C
      DO 130 L=1,K2
      IF (FLG05.GT.0.AND.L.GT.1) GO TO 10
      ND(L)=NN
      M=NN-1
      READ (13) BLANK
      READ (13) BLANK
      READ (13) NN,(TX1(I),I=1,NN)
      READ (13) NN,(TY1(I),I=1,NN)
```

```
      GO TO 20
10    CONTINUE
C
C              * BASIC DATA CALC. AND PRINT (UNTRANSFORMED COORDINATES)
C
      BDN=0
      READ (5,150) ND(2)
      NN=ND(2)
      READ (5,160) (TX1(I),I=1,NN)
      READ (5,160) (TY1(I),I=1,NN)
      GO TO 50
20    SUMS=0.0
      DO 30 I=1,M
      T1=TX1(I+1)-TX1(I)
      T2=TY1(I+1)-TY1(I)
      X2(I)=(TX1(I+1)+TX1(I))/2.
      Y2(I)=(TY1(I+1)+TY1(I))/2.
      DELS(I)=SQRT(T1*T1+T2*T2)
      SUMS=SUMS+DELS(I)
      RSDS(I)=SUMS
30    ALFA(I)=ATAN2(T2,T1)
      MA=M-1
      DO 40 I=1,MA
40    DALF(I)=(ALFA(I+1)-ALFA(I))*57.2957795
50    CONTINUE
      IF (MN) 60,80,60
60    SRM=SQRT(1.-MN*MN)
      DO 70 I=1,NN
70    TX1(I)=TX1(I)/SRM
C
C              * SHIFT X1 AND Y1 TO COMMON /CL/
C
80    IF (BDN) 110,90,110
90    DO 100 I=1,NN
      XP(I)=TX1(I)
100   YP(I)=TY1(I)
      WRITE (12) (XP(I),I=1,NN),(YP(I),I=1,NN)
      GO TO 130
110   DO 120 I=1,NN
      K=K+1
      X1(K)=TX1(I)
120   Y1(K)=TY1(I)
      NT=NT+M
130   CONTINUE
      REWIND 13
C
C              * CALC. PARAMETERS WITH TRANSFORMED COORDINATES AND
C                MACH NO. ADJUSTMENT
C
      J1=0
      N1=ND(1)-1
      DO 140 J=1,N1
      J1=J1+1
      T1=X1(J1+1)-X1(J1)
      T2=Y1(J1+1)-Y1(J1)
      X2(J)=(X1(J1+1)+X1(J1))/2.
      Y2(J)=(Y1(J1+1)+Y1(J1))/2.
      DELS(J)=SQRT(T1*T1+T2*T2)
      COSA(J)=T1/DELS(J)
140   SINA(J)=T2/DELS(J)
      J1=J1+1
C
C              * SAVE PARAMETERS
C
      WRITE (12) (X1(I),I=1,J1),(Y1(I),I=1,J1),(X2(I),I=1,NT),(Y2(I),I=1
     1,NT),(DELS(I),I=1,NT)
```

```
      REWIND 12
C
C               * SAVE SINA AND COSA ON TAPE 4 FOR CALC, OF MATRIX
C                 SOLUTION (RIGHT HAND MATRIX)
      WRITE (4) (SINA(I),I=1,NT),(COSA(I),I=1,NT)
      RETURN
C
C
150   FORMAT (2I5)
160   FORMAT (6F10.0)
      END
      SUBROUTINE MATRIX
C
C               * COMPUTE MATRIX A,B,Z OR X,Y,Z
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ A(200),B(200),AX(200),AY(200),AZ(200),CX(200),CY(200),
     1CZ(200),AXV(200),AYV(200),VN(200,1),VT(200,1),BON,YZERO,IAC,II,JJ,
     2J1,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK,KK
      INTEGER FLG05,BDN
      REAL MN
C
C               * INITIALIZE
      L1=NT
      BON=0.0
      YZERO=0.0
      IAC=1
10    DO 20 I=1,NT
      J=1
      VN(I,J)=0.
20    VT(I,J)=0.
C
C               *  I MIDPOINT LOOP
C
      DO 70 I=1,L1
      II=I
C
C                   J1 IS THE COORDINATE COUNTER
C
      J1=0
      N1=ND(1)-1
      KK=1
      DO 30 J=1,N1
      JJ=J
      J1=J1+1
C
C               * COMPUTE X,Y,Z MATRICES
C
      CALL XYZ
30    CONTINUE
      J1=J1+1
      IF (BON) 40,50,40
C
C               * SAVE X,Y,Z ON TAPE *OFF BODY POINTS
C
40    WRITE (9) (AX(J),J=1,NT),(AY(J),J=1,NT),(AZ(J),J=1,NT)
      GO TO 70
C
C               * SAVE A,B,Z ON TAPE *ON BODY
C
50    DO 60 J=1,NT
      A(J)=-AX(J)*SINA(I)+AY(J)*COSA(I)
```

```
60    B(J)=AX(J)*COSA(I)+AY(J)*SINA(I)
      WRITE (9) (A(J),J=1,NT),(B(J),J=1,NT),(AZ(J),J=1,NT)
70    CONTINUE
C
C               * TEST IF OFF BODY COMPLETED
C               * TEST IF OFF BODY
C
      IF (FLG05.EQ.0.OR.BON.NE.0.) GO TO 90
C
C               * INITIAL FOR OFF BODY * THEN RE-ENTER I,J LOOPS
C
      BON=1.
      L1=ND(2)
      DO 80 I=1,L1
      X2(I)=XP(I)
80    Y2(I)=YP(I)
      GO TO 10
90    REWIND 9
      REWIND 4
      RETURN
      END
      SUBROUTINE XYZ
C
C               * CONTROL FOR X,Y,Z MATRICES COMPUTATION
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ A(200),B(200),AX(200),AY(200),AZ(200),CX(200),CY(200),
     1CZ(200),AXV(200),AYV(200),VN(200,1),VT(200,1),BON,YZERO,IAC,I,J,J1
     2,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK,K
      INTEGER FLG05,BDN
      REAL MN
C
      IF (BON) 50,10,50
10    IF (J-I) 60,20,60
C
C               * J EQUAL I PATH
C
20    T1=.5*DELS(J)
      SJ=T1/Y2(J)
      IF (SJ-.08) 30,30,40
30    CALL XYZ1
      GO TO 190
40    SJ=.08
      CALL XYZ1
      NI=33
      T2=.08*Y2(J)
      DS=(T1-T2)/32.
      DX=DS*COSA(J)
      DY=DS*SINA(J)
      XJ=X2(J)+T2*COSA(J)-DX
      YJ=Y2(J)+T2*SINA(J)-DY
      CALL XYZ2
      GO TO 180
C
C               * INITIAL Y COORDINATE MID-POINT FOR ZERO TEST
C
50    YZERO=Y2(I)-.000001
C
C               * J NOT EQUAL I PATH
C               * COMPUTE MINIMUM DISTANCE TO I MIDPOINT
C
60    D1=(X2(I)-X1(J1))**2+(Y2(I)-Y1(J1))**2
```

```
      D2=(X2(I)-X2(J))**2+(Y2(I)-Y2(J))**2
      D3=(X2(I)-X1(J1+1))**2+(Y2(I)-Y1(J1+1))**2
      IF (D1-D2) 80,80,70
70    IF (D2-D3) 100,100,90
80    IF (D1-D3) 110,110,90
90    DM=SQRT(D3)
      GO TO 120
100   DM=SQRT(D2)
      GO TO 120
110   DM=SQRT(D1)
C
C               * COMPUTE NO. OF INTERVALS(NI) AND DELTA S (DS)
C                 FOR SIMPSON RULE INTEGRATION
C
120   IF (DM.EQ.0.0) GO TO 150
      NI=8.*DELS(J)/DM+0.9
      IF (NI) 130,130,140
130   NI=3
      DS=DELS(J)/2.
      GO TO 170
140   NI=NI+NI
      IF (NI-128) 160,150,150
150   NI=129
      DS=DELS(J)/128.
      GO TO 170
160   XNI=NI
      DS=DELS(J)/XNI
      NI=NI+1
170   DX=DS*COSA(J)
      DY=DS*SINA(J)
180   XJ=X1(J1)-DX
      YJ=Y1(J1)-DY
      CALL XYZ2
190   RETURN
      END
      SUBROUTINE XYZ1
C
C               * COMPUTE X,Y,Z MATRICES FOR SJ LESS THAN OR EQUAL .08
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ A(200),B(200),AX(200),AY(200),AZ(200),CX(200),CY(200),
     1CZ(200),AXV(200),AYV(200),VN(200,1),VT(200,1),BON,YZERO,IAC,I,J,J1
     2,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK,K
      INTEGER FLG05,BDN
      REAL MN
C
C               * INITIALIZE
C
      T1=SJ*SJ
      T2=ALOG(SJ/8.)
      T3=SINA(J)*SINA(J)
      T4=T2+T3
      T5=.666666667*T3
      T6=T5*T3
      T7=SJ+SJ
      T8=T7+T7
      T9=6.2831853*COSA(J)
      T10=6.2831853*SINA(J)
      T11=T1*SJ
C
C               * AXIS FLOW
C
```

```
10    AX(J)=T10+SINA(J)*COSA(J)*(T7+(T4+2.16666667)*T11/12.)
      AX(J)=T10+SINA(J)*COSA(J)*(T7+(T4+2.16666667)*T11/12.)
      AY(J)=T7*T4-T9-(1.+T2-T3-T6)*T11/8.
      T12=T1+T1
      AZ(J)=Y2(J)*T8*(1.-T2+T1*(2.-T12+3.*T2*(1.+T12))/144.)
      RETURN
      END
      SUBROUTINE XYZ2
C
C             * COMPUTE X,Y,Z MATRICES USING SIMPSON RULE INTEGRATION
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /CL/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TL/ A(200),B(200),AX(200),AY(200),AZ(200),CX(200),CY(200),
     1CZ(200),AXV(200),AYV(200),VN(200,1),VT(200,1),BON,YZERO,IAC,I,J,J1
     2,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK,K
      INTEGER FLG05,BDN
      REAL MN
C
C             * INITIALIZE
C
10    S2=.666666667*DS
      S4=S2+S2
      T1=Y2(I)*Y2(I)
C
C             * NO. OF INTERVAL LOOP
C
      DO 130 IS=1,NI
      XJ=XJ+DX
      YJ=YJ+DY
      T2=YJ*YJ
      T3=X2(I)-XJ
      T4=T3*T3
      T5=(Y2(I)+YJ)**2
      T6=T4+T5
      T7=SQRT(T6)
      T8=T2+T4
      T9=(Y2(I)-YJ)**2
      T10=T9+T4
C
C             * COMPUTE ELLIPIC INTEGRAL
C
      XK=4.*YJ*Y2(I)/T6
      CALL ELIP
C
C             * AXIS FLOW
C
      T11=YJ/T7
      IF (Y2(I).EQ.0.) GO TO 20
      T12=YJ/Y2(I)
      FV2=(EKK+EEK*(T1-T8)/T10)/T7
      FV3=Y2(I)/T10*T3/T7*EEK
      F1=FV3*T12
      F2=FV2*T12
      FV4=FV2*T3/Y2(I)
      GO TO 30
20    FV2=0.
      FV3=0.
      FV4=0.
      F2=0.
      F1=T11/T10*T3*EEK
30    F3=T11*EKK
C
```

```
C               * SIMPSON RULE INTEGRATION
C
      IF (IS-1) 40,40,50
C
C               * FIRST PASS
C
40    AXS=F1
      AYS=F2
      AZS=F3
      IA=0
      GO TO 120
50    IF (IS-NI) 60,90,60
60    IF (IA) 80,70,80
C
C               * EVEN PASS
C
70    AXS=AXS+4.*F1
      AYS=AYS+4.*F2
      AZS=AZS+4.*F3
      IA=1
      GO TO 120
C
C               * ODD PASS
C
80    AXS=AXS+F1+F1
      AYS=AYS+F2+F2
      AZS=AZS+F3+F3
      IA=0
      GO TO 120
C
C               * LAST PASS
C
90    IF (J-I) 110,100,110
100   IF (BON.NE.0.0) GO TO 110
      AX(J)=AX(J)-S4*(AXS+F1)
      AY(J)=AY(J)-S2*(AYS+F2)
      AZ(J)=AZ(J)+S4*(AZS+F3)
      GO TO 120
110   AX(J)=-S4*(AXS+F1)
      AY(J)=-S2*(AYS+F2)
      AZ(J)=S4*(AZS+F3)
120   CONTINUE
130   CONTINUE
      RETURN
      END
      SUBROUTINE ELIP
C
C               * HASTINGS APPROXIMATION FOR ELLIPTIC INTEGRALS
C
      COMMON /TL/ A(200),B(200),AX(200),AY(200),AZ(200),CX(200),CY(200),
     1CZ(200),AXV(200),AYV(200),VN(200,1),VT(200,1),BON,YZERO,IAC,I,J,J1
     2,SJ,DS,DX,DY,NI,XJ,YJ,XK,EEK,EKK,K
C
10    ETA=1.-XK
      IF (ETA) 20,20,30
20    WRITE (6,40) ETA
      CALL EXIT
30    ELN=ALOG(ETA)
      EKK=1.3862943611 2+ETA*(0.09666344259+ETA*(0.03590092383+ETA*(0.037
     1425637 13+ETA*0.01451196212)))-ELN*(0.5+ETA*(0.12498593597+ETA*(0.0
     26880248576+ETA*(0.03328355346+ETA*0.00441787012))))
      EEK=1.+ETA*(0.44325141463+ETA*(0.06260601220+ETA*(0.04757383546+ET
     1A*0.01736506451)))-ELN*(ETA*(0.24998368310+ETA*(0.09200180037+ETA*
     2(0.04069697526+ETA*0.00526449639))))
      RETURN
C
```

```
40    FORMAT (1H136H,27H* ERROR IN SUBROUTINE ELIP ,ETA=F15.8)
      END
      OVERLAY(LINK,2,0)
      PROGRAM TWO
C
C              * COMPUTE SOURCE DENSITY SIGMA BY SIEDEL ITERATION
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /C2/ A(200),R(200),NSIG,IT
      DIMENSION ASIG(200,1)
      INTEGER FLG05,BDN
      REAL MN
C
C              * AXIS FLOW
C
      READ (4) (R(I),I=1,NT)
10    REWIND 4
      IT=9
      NSIG=NSIGA
C
C              * SOLVE SIMULTANEOUS EQUATIONS FOR SIGMAS
C
      CALL MISNA2 (ASIG)
      REWIND 9
C
C              * WRITE SIGMAS ON TAPE 3
C
      WRITE (3) (ASIG(I,1),I=1,NT)
      END
      SUBROUTINE MISNA2 (SIG)
C
C              * SOLVE LINEAR SIMULTANEOUS EQUATIONS BY SEIDEL ITERATION
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /C2/ A(200),R(200),NSIG,IT
      DIMENSION SIG(200,1), KFLAG(1), DSIG1(1), DSIG(200,1)
      INTEGER FLG05,BDN
      REAL MN
C
C              * INITIALIZE
C
10    NTU=0
      ITER=0
      NCONV=0
      DO 20 J=1,NSIG
      KFLAG(J)=0
      DO 20 I=1,NT
20    SIG(I,J)=0.0
30    DO 40 I=1,NSIG
40    DSIG1(I)=0.0
C
C              * COMPUTE SIGMA AND DELTA SIGMA
C
      DO 100 I=1,NT
      IF (NTU-3) 50,60,70
C
C              * PLACE A IN LEFT SIDE MATRIX
C
50    READ (9) (A(L),L=1,NT)
C
C              * SAVE LEFT SIDE MATRIX
C
```

```
      WRITE (3) (A(L),L=1,NT)
      WRITE (11) (A(L),L=1,NT)
      GO TO 80
C
C                * READ LEFT SIDE MATRIX
C
60    READ (3) (A(L),L=1,NT)
      GO TO 80
70    READ (11) (A(L),L=1,NT)
80    DO 100 J=1,NSIG
      IF (KFLAG(J).NE.0) GO TO 100
      SUM=0.0
      DO 90 L=1,NT
90    SUM=SUM+A(L)*SIG(L,J)
      DSIG(I,J)=(R(I)-SUM)/A(I)
      SIG(I,J)=SIG(I,J)+DSIG(I,J)
      IF (ABS(DSIG(I,J)).GT.DSIG1(J)) DSIG1(J)=ABS(DSIG(I,J))
100   CONTINUE
C
C                * TEST FOR SOLUTION
C
      REWIND 3
      REWIND 11
      ITER=ITER+1
      DO 110 J=1,NSIG
      IF (KFLAG(J).NE.0) GO TO 110
      IF (DSIG1(J).GE.1.E-6) GO TO 110
      KFLAG(J)=ITER
      NCONV=NCONV+1
      IF (NCONV.EQ.NSIG) GO TO 130
110   CONTINUE
      IF (ITER.EQ.100) GO TO 130
      IF (NTU.EQ.3) GO TO 120
      NTU=3
      GO TO 30
120   NTU=11
      GO TO 30
C
C                * PRINT NO. OF ITERATIONS
C
130   DO 150 J=1,NSIG
      IF (KFLAG(J).NE.0) GO TO 140
      WRITE (6,160) ITERA
      GO TO 150
140   WRITE (6,170) ITERA,KFLAG(J)
150   CONTINUE
      RETURN
C
C
160   FORMAT (1H0,10HFOR ITERA=,I3,46H NO CONVERGENCE IN MISNA2 AFTER 10
     10 ITERATIONS)
170   FORMAT (1H0,10HFOR ITERA=,I3,I5,46H ITERATIONS REQUIRED FOR CONVER
     1GENCE IN MISNA2)
      END
      OVERLAY(LINK,3,0)
      PROGRAM THREE
C
C                * COMPUTE VELOCITY COMPONENTS AND PRINT
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1GS,FLGOS,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /C4/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TC/ RB(200,2),SIG(200,1),A(200),B(200),Z(200),PHI(200,1),X
     1N(200,1),T(200,1),TS(200,1),NSIG,NP,NI,SUMV,SUMM(4)
```

```
      INTEGER FLG05,BDN
      REAL MN
C
      REWIND 3
      IF (FLG05.EQ.0) GO TO 10
C
C              * READ OFF-BODY XP,YP
C
      NP=ND(2)
      READ (12) (XP(I),I=1,NP),(YP(I),I=1,NP)
C
C              * READ X1,Y1,X2,Y2,DELS WITH MACH NO. ADJUSTMENT IF ANY
C
10    NI=NT+1
      READ (12) (X1(I),I=1,NI),(Y1(I),I=1,NI),(X2(I),I=1,NT),(Y2(I),I=1,
     1NT),(DELS(I),I=1,NT)
C
C              * READ SINA,COSA,NO,TO..
C
      READ (4) (A(I),I=1,NT),(B(I),I=1,NT)
      SUMV=0.0
      DO 20 I=1,NT
      SINA(I)=A(I)
      COSA(I)=B(I)
20    SUMV=SUMV+B(I)*DELS(I)*Y2(I)**2
      SUMV=SUMV*3.14159265
      L=1
      DO 30 I=1,NT
      RB(I,L)=A(I)
30    RB(I,L+1)=B(I)
      REWIND 4
      NSIG=NSIGA
      CALL AXIS
      REWIND 13
      BLANK=0.0
      READ (13) DUMMY
      READ (13) DUMMY
      WRITE (13) NN,BLANK,(X2(I),I=1,199)
      WRITE (13) NN,BLANK,(Y2(I),I=1,199)
      WRITE (13) BLANK,(T3(I,1),I=1,199)
      END
      SUBROUTINE AXIS
C
C              * COMPUTE AXISYMMETRIC VELOCITY COMPONENTS AND PRINT
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MN,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /C4/ X1(200),Y1(200),X2(200),Y2(200),DELS(200),SINA(200),CO
     1SA(200),XP(200),YP(200)
      COMMON /TC/ RB(200,2),SIG(200,1),A(200),B(200),Z(200),PHI(200,1),X
     1N(200,1),T(200,1),T3(200,1),INSIG,INP,NI,SUMV,SUMM(4)
      DIMENSION VX(200,1), VY(200,1), VT(200,1), TH(200,1), CP(200,1), S
     1UMTDS(4)
      EQUIVALENCE (VX,XN), (VY,T), (VT,T3), (TH,SIG), (CP,T3)
      REAL MN
      INTEGER FLG05,BDN
C
      NC=NT
      N=1
      NP=INP
C
C              * READ AXIS SIGMAS
C
      SUMM(N)=0.0
      SUMTDS(N)=0.0
```

```
      READ (3) (SIG(I,N),I=1,NC)
C
C               * NO. OF MIDPOINTS LOOP
C
      DO 20 I=1,NT
C
C               * READ MATRICES A,B,Z
C
      READ (9) (A(J),J=1,NT),(B(J),J=1,NT),(Z(J),J=1,NT)
C
C               * NO. OF FLOWS LOOP
C
      N1=0
      N1=N1+2
      SN=0.0
      ST=0.0
      SP=0.0
C
C               * NO. OF ELEMENTS LOOP
C
      DO 10 J=1,NT
      SN=SN+A(J)*SIG(J,N)
      ST=ST+B(J)*SIG(J,N)
10    SP=SP+Z(J)*SIG(J,N)
      XN(I,N)=SN-RB(I,N1-1)
      PHI(I,N)=SP
      T(I,N)=ST+RB(I,N1)
      SUMM(N)=SUMM(N)+PHI(I,N)*Y2(I)*RB(I,N1-1)*DELS(I)
      CP(I,N)=1.-T(I,N)**2
20    CONTINUE
      IF (MN.EQ.0.0) GO TO 60
C
C               * MACH NO. ADJUSTMENT
C
      D1=MN*MN
      D2=1.-D1
      D3=SQRT(D2)
      D4=.7*D1
      D5=.2*D1
      DO 30 I=1,NT
      IF (IMACH.LT.2) BB=D2
      IF (IMACH.GE.2) BB=1.-MN**2*T(I,N)*COSA(I)
      IF (BB.LE.0.0) GO TO 160
      TX=(T(I,N)*COSA(I)-1.)/BB+1.
      TY=T(I,N)*SINA(I)*D3/BB
      T(I,N)=SQRT(TX*TX+TY*TY)
      CP(I,N)=((1.+D5*(1.-T(I,N)**2))**3.5-1.)/D4
30    CONTINUE
      D2=1.-D1
      D3=SQRT(D2)
C
C               * ELIMINATE MACH NO EFFECT FOR PRINTOUT
C
      DO 40 I=1,NI
40    X1(I)=X1(I)*D3
      J1=0
      M=ND(1)-1
      DO 50 J=1,M
      J1=J1+1
      T1=X1(J1+1)-X1(J1)
      T2=Y1(J1+1)-Y1(J1)
      X2(J)=(X1(J1+1)+X1(J1))/2.
      DELS(J)=SQRT(T1*T1+T2*T2)
      COSA(J)=T1/DELS(J)
50    SINA(J)=T2/DELS(J)
      J1=J1+1
```

```
60    CONTINUE
      IF (FLG05.EQ.0) RETURN
C
C               * OFF-BODY POINT
C
      DO 80 I=1,NP
C
C               * READ MATRICES X,Y,Z
C
      READ (9) (A(J),J=1,NT),(B(J),J=1,NT),(Z(J),J=1,NT)
C
C               * NO. OF FLOW
C
      SX=0.0
      SY=0.0
      SP=0.0
C
C               * NO. OF ELEMENTS LOOP
C
      DO 70 J=1,NT
      SX=SX+A(J)*SIG(J,N)
      SY=SY+B(J)*SIG(J,N)
70    SP=SP+Z(J)*SIG(J,N)
      PHI(I,N)=SP
      VX(I,N)=SX+1.
      VY(I,N)=SY
80    CONTINUE
      IF (MN.EQ.0.0) GO TO 110
C               * MACH NO. ADJUSTMENT
      DO 90 I=1,NP
      BB=D2
C
C       LABRUJERE COMPRESSIBILITY CORRECTION
C
      IF (IMACH.GE.2) BB=1.-MN**2*VX(I,N)
      VY(I,N)=VY(I,N)*D3/BB
      VX(I,N)=(VX(I,N)-1.)/BB+1.
90    CONTINUE
      DO 100 I=1,NP
100   XP(I)=XP(I)*D3
C
C               * COMPUTE VT AND THETA
C
110   CONTINUE
      DO 120 I=1,NP
      VT(I,N)=SQRT(VX(I,N)**2+VY(I,N)**2)
120   TH(I,N)=ATAN2(VY(I,N),VX(I,N))*57.2957795
C
C               * PRINT AXIS FLOW (OFF-BODY) OUTPUT
C
      L=1
      I=1
      LCTR=45
130   WRITE (6,170) HEDR
      WRITE (6,180)
      WRITE (6,190)
140   CONTINUE
      CP2=((1.+D5*(1.-VT(I,L)**2))**3.5-1.)/D4
      XM2=VT(I,L)*MN/SQRT(1.-D5*(VT(I,L)**2-1.))
      WRITE (6,210) I,XP(I),YP(I),VX(I,L),VY(I,L),VT(I,L),TH(I,L),XM2,CP
     12
      I=I+1
      IF (I.GT.NP) GO TO 150
      IF (I.LE.LCTR) GO TO 140
      LCTR=LCTR+45
      GO TO 130
```

```
150   CONTINUE
      RETURN
160   WRITE (6,200)
      STOP
C
C
C
170   FORMAT (1H1,25X,23HPOTENTIAL FLOW SOLUTION///6X,8A10//)
180   FORMAT (1X,35H OFF-BODY UNIFORM AXISYMMETRIC FLOW)
190   FORMAT (1X//10X,3HX/L,9X,3HR/L,10X,2HVX,10X,2HVR,10X,2HVT,9X,3HETA
     1,10X,2HML,10X,2HCP//)
200   FORMAT (1X////1X,73HFREESTREAM MACH NUMBER TOO LARGE FOR LABRUJERE
     1 COMPRESSIBILITY CORRECTION/1X,62HRESUBMIT USING IMACH=1 FOR GOETH
     2ERT COMPRESSIBILITY CORRECTION)
210   FORMAT (1X,I3,8F12.6)
      END
      OVERLAY(LINK,5,0)
      PROGRAM FIVE
C
C     VISCOUS FLOW/POTENTIAL FLOW INTERFACE PROGRAM
C
      COMMON HEDR(8),ISWITCH,IPUNCH,ITERA,ITERMAX,IMACH,ISEP,INT(8),IFLA
     1G5,FLG05,MO,APT,ATT,REFL,SREF,XSEPND,NN,BDN,ND(2),NT,NSIGA,IPRINT,
     2AMJET,PTJET,TTJET,RJET,RSTAR
      COMMON /SAVE/ VDUM(402),RDS(201),XIN,VDUM2(347)
      DIMENSION X(200), R(200), CP(200), ME(200), THETA(200), CAPH(200),
     1 CF(200), CAPHI(200), CDF(200), CDP(200), CDT(200), RCPUNCH(200),
     2XO(200), RO(200), CS(200), RI(200), UI(200), DELI(200), RET(200),
     3TAW51(200), PTPT(200), FNN(200), DELTA(200), FLOT(15), FLOTO(7)
      INTEGER FLG05,BDN
      REAL MO,ME
C
C     INITIALIZE
C
      G=1.4
      G1=(G-1.)/2.
      G2=G/(G-1.)
      G3=1./(G-1.)
      G4=(G+1.)/2.
      G5=G4*G3
      G6=G4/G1
      TT=ATT
      PT=APT
      ME(1)=0.0
      THETA(1)=0.0
      CAPH(1)=1.3
      CAPHI(1)=1.3
      CF(1)=0.0
C
C     READ XO,RO,X,R,AND CP FROM TAPE13
C
      REWIND 13
      READ (13) NXO,(XO(I),I=1,NXO)
      READ (13) NXO,(RO(I),I=1,NXO)
      READ (13) NUM,(X(I),I=1,NUM)
      READ (13) NUM,(R(I),I=1,NUM)
      READ (13) CP
C
C     OBTAIN CP AND R AT ORIGINAL X
C
      NTAB=1
      IORDER=2
      IPT=-1
      DO 10 I=2,NUM
      CALL IUNI (200,NUM,X,NTAB,CP,IORDER,XO(I),CS(I),IPT,IERR)
10    CONTINUE
```

```
      DO 20 I=2,NUM
      CP(I)=CS(I)
      R(I)=RO(I)
      X(I)=XO(I)
20    CONTINUE
      REWIND 13
      DO 30 J=1,NUM
      IF (ITERA.EQ.0) RDS(J)=R(J)
      X(J)=X(J)*REFL
      R(J)=R(J)*REFL
      RDS(J)=RDS(J)*REFL
30    CONTINUE
      DO 40 I=1,NXO
      XO(I)=XO(I)*REFL
      RO(I)=RO(I)*REFL
40    CONTINUE
C
C     CALCULATE FREE STREAM CONDITIONS
C
      PO=PT*(1.+G1*MO**2)**(-G2)
      QINF=G/2.*PO*MO**2
      RG=286.96
      PI=3.1415926
      CP(1)=(PT-PO)/(0.5*PO*G*MO**2)
C
C     CALCULATE PLUME BOUNDARY AND VELOCITY
C     USING ONE DIMENSIONAL METHOD
C
      DO 50 I=1,200
      UI(I)=0.0
      RI(I)=0.0
50    CONTINUE
      IT=INT(6)
      NJ=NUM-IT+1
      IF (INT(5).EQ.0) GO TO 90
      RJET=RJET*REFL
      K=0
      DO 80 I=IT,NUM
      K=K+1
      PE=QINF*CP(I)+PO
      PRAT=PTJET/PE
      IF (I.GT.IT) GO TO 70
      PCRIT=(1.0+G1*AMJET**2)**G2
      IF (PRAT.GT.PCRIT) GO TO 60
      RSTAR=PRAT**(-1./G)*SQRT(G4**G6/G1*(1.-PRAT**(-1./G2)))
      RSTAR=SQRT(RSTAR)*RJET
      GO TO 70
60    ASTAR=G4**G5*AMJET/(1.+G1*AMJFT**2)**G5
      RSTAR=SQRT(ASTAR)*RJET
70    XME=SQRT((PRAT**(1./G2)-1.)/G1)
      ASOA=G4**G5*XME/(1.+G1*XME**2)**G5
      IF (I.GT.IT) R(I)=RSTAR/SQRT(ASOA)
      UI(K)=XME*SQRT(G*RG*TTJET/(1.+G1*XME**2))
      RI(K)=R(I)
80    CONTINUE
      RJET=RJET/REFL
90    CONTINUE
C
C     PREPARE INPUT TO SUBROUTINE VISCOUS
C
      INT(1)=NUM
      INT(2)=1
      INT(8)=ISEP-2
      FLOT(1)=12.
      FLOT(2)=0
      FLOT(3)=PT/FLOT(1)**2/47.880258
```

```
      FLOT(4)=TT*1.8
      FLOT(5)=MO
      FLOT(6)=ITERA+1
      FLOT(7)=G
      FLOT(8)=CAPHI(INT(2))
      FLOT(9)=THETA(INT(2))*FLOT(1)/.3048
      FLOT(10)=IPRINT
      FLOT(11)=UI(1)*FLOT(1)/.3048
      FLOT(12)=0.25
      FLOT(13)=0.0
      FLOT(14)=0.0
      FLOT(15)=XSEPND*REFL*FLOT(1)/.3048
      IF (ITERA.EQ.0) XIN=X(NUM)
      IF (INT(7).GT.0) CALL SMINT (X,CP,NUM,INT(3),NUM)
      XIN=XIN*FLOT(1)/.3048
      DO 100 I=1,NUM
      X(I)=X(I)*FLOT(1)/.3048
      R(I)=R(I)*FLOT(1)/.3048
      RDS(I)=RDS(I)*FLOT(1)/.3048
100   CONTINUE
      DO 110 K=1,NJ
      RI(K)=RI(K)*FLOT(1)/.3048
      UI(K)=UI(K)*FLOT(1)/.3048
110   CONTINUE
      CALL VISCUS (INT,FLOT,X,R,CP,RI,UI,FLOTO,RCPUNCH,ME,THETA,DELTA,CA
     1PH,CF,DELI,CAPHI,RET,TAWS1,PTPT,FNN)
C
C     CHANGE OUTPUT FROM VISCOUS TMETERS
C
      FLOTO(7)=FLOTO(7)/FLOT(1)*.3048
      XIN=XIN/FLOT(1)*.3048
      DO 120 I=1,NUM
      X(I)=X(I)/FLOT(1)*.3048
      R(I)=R(I)/FLOT(1)*.3048
      RDS(I)=RDS(I)/FLOT(1)*.3048
      RCPUNCH(I)=RCPUNCH(I)/FLOT(1)*.3048
      THETA(I)=THETA(I)/FLOT(1)*.3048
      DELTA(I)=DELTA(I)/FLOT(1)*.3048
      DELI(I)=DELI(I)/FLOT(1)*.3048
120   CONTINUE
      DO 130 K=1,NJ
      UI(K)=UI(K)/FLOT(1)*.3048
130   CONTINUE
      QS=QINF*SREF
C
C     CALCULATION OF DRAG COEFFICIENTS
C
      CDF(1)=0.0
      CDP(1)=0.0
      CDT(1)=0.0
      ROLD=0.0
      QOLD=0.0
      DO 140 J=2,NUM
      PE=PO*(1.+G/2.*MO**2*CP(J))
      QNEW=G/2.*PE*ME(J)**2
      RNEW=RO(J)
      ANGLE=ATAN((RNEW-ROLD)/(X(J)-X(J-1)))
      SL=PI*(RNEW+ROLD)*SQRT((RNEW-ROLD)**2+(X(J)-X(J-1))**2)
      CDF(J)=(CF(J)+CF(J-1))*(QNEW+QOLD)*SL*COS(ANGLE)/QS/4.+CDF(J-1)
      CDP(J)=PI/SREF*(RNEW*CP(J)+ROLD*CP(J-1))*(RNEW-ROLD)+CDP(J-1)
      CDT(J)=CDP(J)+CDF(J)
      ROLD=RNEW
      QOLD=QNEW
140   CONTINUE
C
C     OUTPUT   DATA
```

```
C
      DO 150 N=1,NUM
      X(N)=X(N)/REFL
      R(N)=R(N)/REFL
      THETA(N)=THETA(N)/REFL
      DELTA(N)=DELTA(N)/REFL
      RCPUNCH(N)=RCPUNCH(N)/REFL
      RDS(N)=RDS(N)/REFL
      DELI(N)=DELI(N)/REFL
150   CONTINUE
      IF (ITERA.LT.IPRINT) GO TO 170
      FLOTO(7)=FLOTO(7)/REFL
      XINND=XIN/REFL
      N1=1
160   N2=N1+34
      IF (N2.GE.NUM) N2=NUM
      WRITE (6,200) HEDR,ITERA,MO,TT,PT,REFL,SREF
      PRINT 210, FLOTO(7),XINND
      WRITE (6,220)
      WRITE (6,230) (X(N),R(N),CP(N),CF(N),CDP(N),CDF(N),CDT(N),RDS(N),R
     1CPUNCH(N),DELTA(N),DELI(N),THETA(N),CAPH(N),N=N1,N2)
      IF (N2.GE.NUM) GO TO 170
      N1=N2+1
      GO TO 160
170   CONTINUE
      REWIND 13
      READ (13) BLANK
      READ (13) BLANK
      NN=NUM
      WRITE (13) NN,(X(I),I=1,NN)
      WRITE (13) NN,(RCPUNCH(I),I=1,NN)
      WRITE (13) CP
C
C     WRITE DATA ON TAPE 7 FOR RESTART
C
      IF (IPUNCH.LT.1) GO TO 190
      IF (ITERA.NE.ITERMAX) GO TO 190
      DO 180 I=1,NXO
      XO(I)=XO(I)/REFL
      RO(I)=RO(I)/REFL
180   CONTINUE
      IDM=ITERMAX+1
      REWIND 7
      WRITE (7,270) HEDR
      WRITE (7,240) ISWITCH,IPRINT,IPUNCH,IDM,IDM,IMACH,ISEP,(INT(I),I=2
     1,7),IFLAGS
      WRITE (7,280) MO,APT,ATT,REFL,SREF,XSEPND
      WRITE (7,280) AMJET,PTJET,TTJET,RJET
      WRITE (7,240) NN
      WRITE (7,290) (XO(I),I=1,NN)
      WRITE (7,300) (RO(I),I=1,NN)
      WRITE (7,250) (RCPUNCH(I),I=1,NN)
      WRITE (7,260) (CP(I),I=1,NN)
190   CONTINUE
C
C
200   FORMAT (1H1,8A10,5X,14HITERATION NO  ,I2//2X,4HMO =,F7.4,4X,4HTT =
     1,F7.2,7H KELVIN,4X,4HPT =,F10.1,8H PASCALS,4X,3HL =,F10.6,7H METER
     2S,4X,6HSREF =,F10.6,10H SQ METERS//)
210   FORMAT (4X,34HBOUNDARY LAYER SEPARATION AT X/L =,F10.6,12X,36HBOUN
     1DARY LAYER REATTACHMENT AT X/L =,F10.6,//)
220   FORMAT (5X,3HX/L,5X,3HR/L,5X,2HCP,6X,2HCF,6X,3HCDP,5X,3HCDF,5X,3HC
     1DT,4X,5HRDS/L,4X,4HRC/L,2X,6HDEL*/L,3X,5HDEL/L,2X,7HTHETA/L,3X,1HH
     2/)
230   FORMAT (1X,13F8.4)
240   FORMAT (16I5)
```

```
250   FORMAT (6F10.6,4X,2HRC)
260   FORMAT (6F10.6,4X,2HCP)
270   FORMAT (8A10)
280   FORMAT (F10.6,F10.1,F10.2,3F10.6)
290   FORMAT (6F10.6,4X,2HXO)
300   FORMAT (6F10.6,4X,2HRO)
      END
      SUBROUTINE VISCUS (INT,FLOT,XA,RAD,CP,RI,UJ,FLOTO,RADO,A,THR,DEL51
     1,H51,CFA,DEL1,H1,RET,TAW51,PTB51,FNN51)
C
C          VISCOUS  FLOW  SUBROUTINE  PACKAGE
C
      COMMON /SAVE/ SB(201),SC(201),Y(201),XIN,XSEPSV(20),DELSV(20),YOUT
     1(201)
      DIMENSION INT(8), FLOT(15), FLOTO(7), XA(201), RAD(201), U(201), R
     1ADO(201), A(201), THR(201), DEL51(201), H51(201), CFA(201), DEL1(2
     201), H1(201), RET(201), TAW51(201), PTB51(201), FNN51(201), CP(201
     3), RI(201), UJ(201), VBLC(201), SS(201), S1(201), DSTAR(201), CSV(
     4201), CPCV(4)
C
      ANA=FLOT(6)
      IAN=ANA
      THR(1)=0.
      DEL51(1)=0.
      DEL1(1)=0.
      H51(1)=0.
      RET(1)=0.
      TAW51(1)=0.
      PTB51(1)=0.
      FNN51(1)=0.
      NN=INT(1)
      NAZ=INT(2)
      NMIN=INT(3)
      NMAX=INT(4)
      IJET=INT(5)
      NEXT=INT(6)
      ISMOD=INT(7)
      IPRESS=INT(8)
      Z=FLOT(1)
      TWW=FLOT(2)
      PT=FLOT(3)
      TT=FLOT(4)
      AMIN=FLOT(5)
      GA=FLOT(7)
      HIX=FLOT(8)
      THRR=FLOT(9)
      C=FLOT(12)
      XSIN=FLOT(15)
      IF (IAN.EQ.1) XIN=XA(NN)
      R=53.35
      GC=32.174
      PFREE=PT*(1.+(GA-1.)*.5*AMIN**2)**(GA/(1.-GA))
      IF (ANA.GT.1) GO TO 20
      DO 10 I=1,NN
10    Y(I)=RAD(I)
      XSEP=0.
20    CONTINUE
      DO 30 I=1,NN
30    SB(I)=3.1416*Y(I)**2
C
C          CALCULATE  VELOCITY  FROM  CP
C
      DO 40 I=1,NN
      PL=.5*GA*PFREE*AMIN**2*CP(I)+PFREE
      AML2=2./(GA-1.)*((PL/PT)**((1.-GA)/GA)-1.)
      IF(AML2.LE.0.0)  AML2=0.000000001
```

```
      AML=SQRT(AML2)
      TL=TT/(1.+(GA-1.)*.5*AML**2)
40    U(I)=SQRT(2./(GA-1.)*GA*R*GC*(TT-TL))
C
C          SHAPEJ  CALCULATES  1ST  DERIVATIVE  OF  CONTOUR
C
      CALL SHAPEJ (SS,S1,XA,NN)
C
C          NEWBL  CONTROL  CALCULATION  OF  BOUNDARY  LAYER
C
      CALL NEWBL (VBLC,XA,Y,SS,NAZ,NN,TWW,Z,PT,TT,ANA,GA,U,S1,CFA,HIX,TH
     1RR,A,DEL1,RET,THR,DSTAR,DEL51,H51,TAW51,PTB51,FNN51,H1,DRAG)
      FLOTO(1)=DRAG
      IF (IPRESS.GE.0) GO TO 50
      IF (IPRESS.EQ.-1.AND.XSIN.NE.0.0) GO TO 90
      XSIN=XA(NEXT-1)
      GO TO 85
50    CONTINUE
C
C          FIX  DETERMINES  MIN.  CP.  THE  MIN  CP  IS  USED  AS  START
C               LOCATION  IN  SEARCH  FOR  SEPARATION
C
      CALL FIX (NMIN,NMAX,CP,MI)
      CPS=0
      CPCV(4)=1.00
C
C          SEPA  DETERMINES  SEPARATION  PROPERTIES
C
      IF (MI-NN) 60,170,170
60    CONTINUE
      MM=MI-NAZ
      CALL SEPA (XA,RAD,CP,AMIN,CFA(MI),DEL1(MM),THR(MM),RET(MM),CP(MI),
     1MI,NMAX,CPCV)
      CPS=CPCV(IPRESS+1)
      FLOTO(2)=CPCV(1)
      FLOTO(3)=CPCV(2)
      FLOTO(4)=CPCV(3)
      FLOTO(5)=CPCV(4)
      DO 80 I=MI,NMAX
      IF (CP(I)-CPS) 80,70,70
70    XSEP=((CPS-CP(I-1))/(CP(I)-CP(I-1)))*(XA(I)-XA(I-1))+XA(I-1)
      AML=A(I)
      GO TO 90
80    CONTINUE
      IF (IAN.EQ.1) GO TO 90
      IF (XSIN.NE.0.) GO TO 90
      IF (XSEPSV(IAN-1).EQ.0.) GO TO 90
85    XSEP=XA(NEXT-1)
      WRITE (6,220) XSEP
90    CONTINUE
C
C     CALCULATE SEPARATION POINT
C
      IF (ANA.EQ.1.) GO TO 160
      IF (ANA.GT.2.) GO TO 100
      DELSV(2)=ABS(XSEP-XSEPSV(1))
      XSEP=XSEPSV(1)
      GO TO 160
100   CONTINUE
      IF (ANA.EQ.3.) GO TO 140
      IF (ANA.GE.8.) GO TO 150
      AVEDEL=0.
      IAN1=IAN-1
      DO 110 IBJ=2,IAN1
110   AVEDEL=AVEDEL+DELSV(IBJ)
      AVEDEL=AVEDEL/IAN1
```

```
      IF (ABS(XSEP-XSEPSV(IAN-1)).LT.2.*AVEDEL) GO TO 140
      IF (XSEP-XSEPSV(IAN-1)) 120,120,130
120   XSEP=XSEPSV(IAN-1)-2.*AVEDEL
      GO TO 140
130   XSEP=XSEPSV(IAN-1)+2.*AVEDEL
140   DELSV(IAN)=ABS(XSEP-XSEPSV(IAN-1))
      XSEP=(XSEP+XSEPSV(IAN-1))*.5
      GO TO 160
150   CONTINUE
      XSEP=AMIN1(XSEPSV(7),XSEPSV(6),XSEPSV(5))
160   XSEPSV(IAN)=XSEP
170   CONTINUE
      DO 180 I=1,NN
180   CSV(I)=CP(I)
      IF (XSIN.NE.0.) XSEP=XSIN
      IF (ANA.GT.4) GO TO 200
C
C          ZERO  CP  FOR  THE  FIRST  4  ITERATIONS
C
      DO 190 I=1,NN
      RATIO=0.
190   CSV(I)=CP(I)*RATIO
C
C          SEP   CALCULATES  THE  AERODYNAMIC  CONTOUR
C
200   CALL SEP (NN,XA,RAD,CSV,XSEP,AMIN,GA,TT,PT,RADO,DSTAR,Y,ANA,IJET,N
     1EXT,RI,UJ,C)
      FLOTO(7)=XSEP
      IF (ISMOO.EQ.0) GO TO 210
      CALL SMINT (XA,RADO,NN,NMIN,NMAX)
C     WRITE (6,220) (I,XA(I),RADO(I),I=1,NN)
210   RETURN
C
C
C
220   FORMAT (54H DID NOT SEPARATE, USE NOZZLE EXIT AS SEPARATION POINT,
     1E12.4)
      END
      SUBROUTINE SHAPEJ (SS,S1,X,NN)
C
C     THIS SUBROUTINE SETS THE BOUNDARY CONDITIONS. THESE BOUNDARY
C          CONDITIONS ARE SET BY THE INITIAL AND FINAL SLOPES OF THE
C          CROSSECTIONAL AREA CURVES.
C
      DIMENSION C1SS(201), C2SS(201), C3SS(201), SS(1), S1(1), X(1)
C
      CALL POWER (X,NN)
      CALL SUMA (2,NN-1,3,X,SS,C1SS,C2SS,C3SS,S1,1)
      S1(1)=0.0
      S1(NN)=0.0
      RETURN
      END
      SUBROUTINE POWER (X,NN)
C
      DIMENSION X(1)
      COMMON /COEFF/ X2(201),X3(201),X4(201)
C
      DO 10 I=1,NN
      X2(I)=X(I)*X(I)
      X3(I)=X2(I)*X(I)
      X4(I)=X3(I)*X(I)
10    CONTINUE
      RETURN
      END
      SUBROUTINE SUMA (NX,NY,LZ,X,S,C1,C2,C3,S1,L)
C
```

```
C     THIS SUBROUTINE CURVE FITS A PARABOLIC ARC THRU LEAST SQUARES
C
      COMMON /COEFF/ X2(201),X3(201),X4(201)
      DIMENSION X(1), S(1), S1(1), C1(1), C2(1), C3(1)
      DOUBLE PRECISION SUM1,SUM2,SUM3,SUM4,SUM5,SUM6,SUM7
C
      LN=LZ/2
      C1(NX-1)=0.0
      C2(NX-1)=0.0
      C3(NX-1)=0.0
      DO 30 J=NX,NY
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      SUM4=0.0
      SUM5=0.0
      SUM6=0.0
      SUM7=0.0
      M=J-LN
      MM=J+LN
      DO 10 I=M,MM
      SUM1=SUM1+X(I)
      SUM2=SUM2+X2(I)
      SUM3=SUM3+X3(I)
      SUM4=SUM4+X4(I)
      SUM5=SUM5+(X(I)*S(I))
      SUM6=SUM6+(X2(I)*S(I))
      SUM7=SUM7+S(I)
10    CONTINUE
      AA=SUM7
      AB=SUM1
      AC=SUM2
      AD=SUM5/SUM1
      AE=SUM2/SUM1
      AF=SUM6/SUM2
      AG=SUM3/SUM1
      AH=SUM4/SUM2
      AI=SUM3/SUM2
      AAAR=AA/LZ
      A=AAAR-AD
      C=AAAR-AF
      ABAR=AB/LZ
      B=ABAR-AE
      D=ABAR-AI
      ACAR=AC/LZ
      E=ACAR-AG
      G=ACAR-AH
      B=B+.1D-10
      D=D+.1D-10
      EB=E/B
      AM=EB-G/D
      IF (ABS(AM).LE.0.1D-10) GO TO 20
      AOB=A/B
      C3(J)=(AOB-C/D)/AM
      C2(J)=AOB-EB*C3(J)
      C1(J)=AAAR-C2(J)*ABAR-C3(J)*ACAR
      GO TO 30
20    CONTINUE
      C3(J)=C3(J-1)
      C2(J)=C2(J-1)
      C1(J)=C1(J-1)
30    CONTINUE
      IF (L.EQ.0) RETURN
C
C     COMPUTE 1ST DERIV. OF X VS S CURVE.
C
```

```
      DO 40 J=NX,NY
      S1(J)=C2(J)+2*C3(J)*X(J)
40    CONTINUE
      DO 50 J=1,LN
      K=NX-LN+J-1
      S1(K)=C2(NX)+2*C3(NX)*X(K)
      I=J+NY
      S1(I)=C2(NY)+2*C3(NY)*X(I)
50    CONTINUE
      RETURN
      END
      SUBROUTINE NEWBL (VBLC,X,YO,S,NAZ,NN,TWW,Z,PT,TT,ANA,GA,U,S1,CFA,H
     1IX,THRR,AM,DEL1,RET,THR,DSTAR,DEL51,H51,TAW51,PTB51,FNN51,H1,DRAG)
C
      DIMENSION H1(1), DSTAR(201), U(1), VBLC(1), X(1), YO(1), S(1), S1(
     11), CFA(201), AM(1), THR(1), DEL1(1), RET(1), DEL51(1), H51(1), TA
     2W51(1), PTB51(1), FNN51(1), BC(201), CPB(201), YBAR(201)
      COMMON /SAVE/ SB(201),SC(201),Y(201),XIN
C
      NBJ=NN-NAZ+1
      DO 10 I=1,NN
10    VBLC(I)=U(I)
      DO 20 KJ=1,NBJ
      NJA=NAZ+KJ-1
      CPB(KJ)=VBLC(NJA)
      YBAR(KJ)=X(NJA)
      BC(KJ)=YO(NJA)
20    CONTINUE
      ABC=PT
      CALL BLC (PT,TT,YBAR,BC,CPB,TWW,Z,NBJ,DSTAR,THRR,HIX,H1,CFA,AM,GA,
     1DEL1,RET,THR,DEL51,H51,TAW51,PTB51,FNN51,DRAG)
      PT=ABC
      IF (NAZ) 60,60,30
30    DO 40 NJ=1,NBJ
      NAJ=NAZ+NJ-1
      CPB(NAJ)=CFA(NJ)
      BC(NAJ)=AM(NJ)
      YBAR(NAJ)=DSTAR(NJ)
40    CONTINUE
      DO 50 NJ=1,NAZ
      YBAR(NJ)=DSTAR(2)
      CPB(NJ)=CFA(2)
      BC(NJ)=AM(1)
50    CONTINUE
60    CONTINUE
      DO 100 I=1,NN
      CFA(I)=CPB(I)
      AM(I)=BC(I)
      DSTAR(I)=YBAR(I)
      IF (S(I)-.1E-8) 70,70,80
70    RCO=0.0
      GO TO 90
80    CONTINUE
      RCO=S1(I)/(2.0*SQRT(3.1416*S(I)))
      RCO=ABS(RCO)
90    CONTINUE
      DEV=RCO**2+1.0
      SUQ=1.0/DEV
      AMB=SQRT(SUQ)
      DSTAR(I)=DSTAR(I)/AMB
100   CONTINUE
      IF (ANA.GT.1.) GO TO 120
      DO 110 I=1,NN
      SC(I)=0.
110   SB(I)=0.
120   ABC=0.
```

```
      DO 140 I=1,NN
      IF (ANA.LE.3.) GO TO 130
      DSTAR(I)=.25*DSTAR(I)+.5*SB(I)+.25*SC(I)
130   SC(I)=SB(I)
      SB(I)=DSTAR(I)
140   CONTINUE
      RETURN
      END
      SUBROUTINE BLC (PT,TT,XV,YV,V,TWW,Z,NN,DSTAR,THRR,HIX,HICH,CFA,AM,
     1GAM,DEL1,RET,THB,DEL51,H51,TAW51,PTB51,FNN51,DRAG)
C
      DIMENSION AM(1), XV(1), YV(1), V(1), DSTAR(1), HICH(1), DEL1(1), C
     1FA(201), DEL51(1), H51(1), TAW51(1), PTB51(1), FNN51(1), RET(1), T
     2HB(1), X(201), Y(201)
C
      TF(X)=1.+.2*X**2
      PF(X)=TF(X)**3.5
      TAWE(X)=1.+.178*X**2
      H2(X)=(X*(X+1.)**2)/2.
      H3(HI)=2.*HIF/(HI+1.)*(HI-1.)-.5*((HI+1.)*.3/4.3)
C
      G1=(GAM-1.)/2.
      G2=GAM/(1.-GAM)
      DSTAR(1)=HIX*THRR
      THR=THRR
      IF (THR.LT..00001) THR=.00001
      DO 10 I=1,NN
      X(I)=XV(I)/Z
      AM(I)=V(I)/49./SQRT(TT-V(I)**2/.48/778./32.17)
10    Y(I)=YV(I)/Z
      IF (AM(1).LE.0.00001) AM(1)=AM(2)
      PT=PT*Z*Z
      L=1
      HIF=1.3
      U=2.27E-08*TT**1.5/(TT+198.6)
      AU=SQRT(1.4/1716./TT)*PT/U
      M=NN-1
      THTR=THR/TF(AM(1))**3/Z
      HM=1.
      DRAG=0.0
      THT=0.
      HD=0.
      IF (HIX) 20,20,30
20    HI=1.3
      GO TO 40
30    HI=HIX
40    DO 230 I=1,M
      DM=AM(I+1)-AM(I)
      DY=Y(I+1)-Y(I)
      DX=X(I+1)-X(I)
      XXN=N
      DLM=ABS(DM/AM(I)+DM/AM(I+1))+.001*DX/THR*Z+XXN*HD
      IF (Y(I+1)) 50,60,50
50    DLM=DLM+ABS(DY/Y(I+1))
60    N=30.*DLM
      IF (N-10) 70,70,80
70    N=10
80    IF (30-N) 90,100,100
90    N=30
100   S=N
      DX=DX/S
      YY=Y(I)-DY/2./S
      DY=DY/S
      AA=AM(I)-DM/2./S
      DM=DM/S
      DL=SQRT(DX**2+DY**2)
```

```
      N=8
      DO 220 J=1,N
      YY=YY+DY
      AA=AA+DM
      TE=TT/TF(AA)
      TAW=TE*TAWE(AA)
      IF (TWW) 110,110,120
110   TW=TAW
      GO TO 130
120   TW=TWW
130   TR=(TAW+TE)/2.+.22*TE*(TAWE(AA)-1.)
      TR=TR+(TW-TAW)/2.
      THH=THTR
      HH=HI
      THTR=(THH+THT/2.)
      HI=HD/2.+HH
      AY=ABS(THTR)
      A=.123*EXP(-1.561*HI)*(AA*AU*AY)**(-.268)*(TE/TR)**.732*(TE/TT)**3
     1.268
      A=A*(TT/TR)**.0645
      THT=+A*DL-THTR*DM/AA*(2.+HI+(TW/TT-1.)*HIF/HM)
      THT=(THTR*(-DY/YY)+THT)
      THTR=(THH+THT/2.)
      HD=-DM/AA*H2(HI)*(HI-1.+(TW/TT-1.)*H3(HI)/HM)
      HF=(HI-(HI+1.)*.36*(EXP(2.9*(HI-1.))-1./HI))
      HF=HF*(HI**2-1.+EXP(20.-20.*HI)*.1)
      HD=HD+HF/THTR*A*DL
      IF (ABS(HD)/HI-.2) 150,150,140
140   HD=.2*HD/ABS(HD)*HI
150   HI=HH+HD
      THTR=(THH+THT)
      IF (HI) 160,160,170
160   HI=.5
      HD=0.
      GO TO 190
170   CONT=2.0
      IF (HI-CONT) 190,190,180
180   HI=CONT
      HH=CONT
      HD=0.
190   TFA=TF(AA)
      THR=THTR*TFA**3
      CF=2.*A*TFA**3
      IF (J+I-2) 200,200,210
200   RV=PT/PF(AA)*SQRT(1.4/TE/1716.)*AA
210   HEAD=0.7*(AA*AA)*PT
      HEAD=HEAD/((1.0+(0.2*(AA*AA)))**3.5)
      CFQ=CF*HEAD
      DRAG=DRAG+(6.2832*CFQ*YY*DX)
      RV=PT/PF(AA)*SQRT(1.4/TE/1716.)*AA
      HTR=HI+(TW/TT-1.)*HIF/HM
      H=HTR*TFA+.2*AA**2
      U=2.27E-08*TE**1.5/(TE+198.6)
      RE=RV*THR/U
      THR=THR*Z
      DEL=THR*H
      DSTAR(I+1)=DEL
220   CONTINUE
      FNN=2.0/(HI-1.)
      FM1=1.+G1*(AM(I+1)**2)
      FM2=FM1-1.0
      FM3=GAM*(AM(I+1)**2)/2.
      TTT=THR/FM1**3
      DDD=TTT*(1.+FNN)*(2.+FNN)/FNN
      DSTA=DEL/FM1**3
      AGB=TTT+DSTA-DDD
```

```
      DEL1(I+1)=DDD*(FM1**4)+AGB*FM2*(FM1**3)
      PTBAR=FM1**G2*(1.+FM3*(1.-(DEL+THR)/DEL1(I+1)))
      L=L+1
      DEL51(L)=DEL
      H51(L)=H
      TAW51(L)=TAW
      PTB51(L)=PTBAR
      FNN51(L)=FNN
      CFA(I+1)=CF
      HICH(I+1)=HI
      RET(I+1)=RE
      THB(I+1)=THR
230   CONTINUE
      CFA(1)=CFA(2)
      RETURN
      END
      SUBROUTINE FIX (NMIN,NN,CP,MI)
C
      DIMENSION CP(1)
C
      MI=NMIN
      PMIN=CP(NMIN)
      DO 20 I=NMIN,NN
      IF (CP(I)-PMIN) 10,20,20
10    PMIN=CP(I)
      MI=I
20    CONTINUE
      RETURN
      END
      SUBROUTINE SEPA (X,R,CP,EMI,CF1,DEL1,THETA1,RETH1,CP1,NSN,NEN,CPRT
     1)
C
      DIMENSION X(201), R(201), CP(201), CPD(201), C1(201), C2(201), C3(
     1201), CPRT(4), YY(201), PSP1(201), E(201)
      COMMON /COEFF/ X2(201),X3(201),X4(201)
      COMMON /BCB/ TTRAT(201),PTRAT(201),PTRNS(201),URAT(201),EM(201),WR
     1(201),PHIR(201),TWTTE,GAMMA,BLMN,BLMON,VWVE1,C,DSD,DDSD,BK,EME1,SH
     2FAC,SIG1,SIGMA1,SIGS1
C
      ASIN(X)=ATAN(X/SQRT(1-X*X))
C
      EM1=0.0
      BTAIL=0.0
      ELLBT=0.0
      ENTMAS=0.0
      ENPRES=3.0
      CFWCF1=1.0
      GAMMA=1.4
      GAM=GAMMA
      EM1=SQRT(5.*((1.+.2*EMI*EMI)*(1.+.7*EMI*EMI*CP1)**((1.-GAM)/GAM)-1
     1.))
      EME1=EM1
      EM1EMI=EM1/EMI
      CPSG=CP1+200.*CF1*(1.+(GAM/2.)*EMI*EMI*CP1)*EM1EMI*EM1EMI
C
C     CALCULATE CP(SEP) AND P(SEP)/PI USING GOLDSCHMEID METHOD
C
      PSPIG=1.+.5*GAM*CPSG*EMI*EMI
      PSPIG=PSPIG/(1.+.5*GAM*CP1*EMI*EMI)
C
C     STRATFORDS SEPARATION CALCULATION
C
      NX=NSN+2
      NY=NEN-2
      LZ=5
      L=1
```

```
      CALL SUMA (NX,NY,LZ,X,CP,C1,C2,C3,CPD,L)
      RE=RETH1*(X(NSN)/THETA1)
      SS=DEL1/(.37*RE**(-.2))
      RHS=.39*(10.**(-6.)*RE)**(+.1)
      DIF=0
      JK=0
      DO 100 I=NSN,NEN
      DX=X(I)-X(NSN)
      S=SS+DX
      IF (CPD(I)) 90,90,10
10    CONTINUE
      HLS=CP(I)*(S*CPD(I))**.5
      DIF=RHS-HLS
      IF (JK-1) 20,40,40
20    CONTINUE
      JK=1
      LNZ=0
      IF (DIF) 30,30,40
30    LNZ=1
40    CONTINUE
      IF (LNZ) 50,50,70
50    IF (DIF) 60,60,90
60    IA=I
      GO TO 110
70    IF (DIF) 90,80,80
80    IA=I
      GO TO 110
90    CONTINUE
100   CONTINUE
      GO TO 120
110   CONTINUE
      CPSS=CP(IA-1)
      CPRT(4)=CPSS
120   CONTINUE
C
C     CALCULATE CP(SEP) AND P(SEP)/PI USING MODIFIED PAGE METHOD
C
      CPSP=CP1+0.38*(1.+(GAM/2.)*EM1*EM1*CP1)*EM1EM1*EM1EM1
C
C     EVALUATE PROFILE PARAMETERS AT STATION 1
C
      BK=0.4
      C=5.1
      TWTTE=1.
      SIG1=0.2*EM1*EM1
      SIGMA1=SIG1/(1.+SIG1)
      SIGS1=SQRT(SIGMA1)
      FSIG1=(ASIN(SIGS1))/SIGS1
      VWVE1=(1.+SIG1)**1.76
      UTUES1=SQRT((CF1/2.)*(SIGMA1/(1.-SIGMA1)))/ASIN(SQRT(SIGMA1))
      REDEL1=RETH1*DEL1/THETA1
      PX1=.5*(1.-UTUES1*((1./BK)*ALOG(REDEL1*ABS(UTUES1)*FSIG1/VWVE1)+C)
     1)
C
C     DETERMINE B.L. PROFILE PROPERTIES AT STATION 1
C
      CALL PRFL (UTUES1,PX1,EM1,1.0,2,YY)
C
C     DETERMINE UPSTREAM BOUNDARY LAYER INTEGRAL PROPERTIES
C
      CALL FLUX (101,YY,EM1)
      RETH1=REDEL1*DDSD
      AMASS1=BLMN
      AMOM1=BLMON
C
C     START SOLUTION PROCEDURE, ASSUME PS/P1
```

```
      DOD=DSD
      DDOD=DDSD
      UTUESS=0.
      PSP1(1)=PSP1G
      P1PT=(1.+.2*EM1*EM1)**(GAM/(1.-GAM))
      FEMS1=(PSP1(1)*P1PT)**((1.-GAM)/GAM)-1.
      FEMS2=P1PT**((1.-GAM)/GAM)-1.
      EMS=EM1*SQRT(FEMS1/FEMS2)
      PXS=0.5
      J=1
      GO TO 140
130   J=J+1
      IF (J.EQ.80) GO TO 290
      PSP1(J)=PSP1(J-1)-0.1
      FEMS1=(PSP1(J)*P1PT)**((1.-GAM)/GAM)-1.
      FEMS2=P1PT**((1.-GAM)/GAM)-1.
      EMS=EM1*SQRT(FEMS1/FEMS2)
      PXS=0.5
C
C     ENTRAINMENT AND FRICTION CONSTANTS
C
      IF (ENTMAS.GT.0.) GO TO 270
C
C     CALCULATE ENTRAINMENT FROM GREENS THEOREM
C
140   ELDEL1=ELLBT*BTAIL/DEL1
      AP1PI=1.0+.7*EMI*EMI*CP1
      PGX=(1.0+.7*EMI*EMI*CP(NSN-1))/AP1PI
      IJB=NEN
      DO 230 II=NSN,NEN
      PGX1=PGX
      PGX=(1.0+.7*EMI*EMI*CP(II))/AP1PI
      IF (II-NSN) 150,150,160
150   PGX1=PGX
      GO TO 230
160   CONTINUE
      DPG=PGX-PSP1(J)
      IF (DPG) 190,170,170
170   IF (PGX1-PSP1(J)) 180,220,220
180   DBG=ABS(PSP1(J)-PGX1)
      DTG=ABS(PGX-PGX1)
      GO TO 210
190   IF (PGX1-PSP1(J)) 220,220,200
200   DBG=ABS(PSP1(J)-PGX1)
      DTG=ABS(PGX-PGX1)
210   AL=X(II-1)-X(NSN)+(DBG/DTG)*(X(II)-X(II-1))
      IJB=II-1
      GO TO 240
220   CONTINUE
230   CONTINUE
      AL=X(NEN)-X(NSN)
240   ELDEL1=AL/DEL1
      IF (J.LE.2) GO TO 260
      TOTAL=0.
      DO 250 II=NSN,IJB
250   TOTAL=TOTAL+CP(II)
      CPAVE=TOTAL/(IJB-NSN+1)
      CPCV=(PSP1(J)*(1.+.7*EMI*EMI*CP1)-1.)/(.7*EMI*EMI)
      ENPRES=(CPCV-CP1)/(CPAVE-CP1)
260   CONTINUE
      FENT1=(1.-DOD)/DDOD-3.
      FENT2=FENT1**(-0.6169)
      FENT3=1.-DOD
      AMEMBL=ELDEL1*.0299*FENT2/FENT3
      GO TO 280
270   AMEMBL=ENTMAS
```

```
280   CONTINUE
      CALL PRFL (UTUESS,PXS,EMS,PSP1(J),1,YY)
      CALL FLUX (101,YY,EMS)
      AMASSS=BLMN
      AMOMS=BLMON
      ALHS1=(1./PSP1(J))*(1.+AMEMBL)*EM1/EMS
      ALHS2=SQRT((1.+.2*EM1*EM1)/(1.+.2*EMS*EMS))
      ALHS3=AMASS1/AMASSS
      ALHS=ALHS1*ALHS2*ALHS3
      RHS1=(1./ENPRES)*(PSP1(J)-1.)
      RHS2=1.4*EM1*EM1*AMOM1
      RHS3=1.4*AMEMBL*EM1*EM1*AMASS1
      RHS4=.5*CF1*CFWCF1*.70*EM1*EM1*ELDEL1
      RHS5=1.+(1./ENPRES)*(PSP1(J)-1.)-PSP1(J)
      RHS6=PSP1(J)*1.4*EMS*EMS*AMOMS
      RHS=(RHS1-RHS2-RHS3+RHS4)/(RHS5-RHS6)
      E(J)=RHS-ALHS
      TEST=ABS(E(J))
      IF (TEST.LE.0.00001) GO TO 300
      IF (J.EQ.1) GO TO 130
      IF (ABS(E(J)-E(J-1)).LE..001*ABS((E(J)+E(J-1))*.5)) GO TO 300
      SLOPE=(E(J-1)-E(J))/(PSP1(J-1)-PSP1(J))
      PSP1(J+1)=PSP1(J)-E(J)/SLOPE
      IF (PSP1(J+1).LT.0.) GO TO 130
      FEMS1=(PSP1(J+1)*P1PT)**((1.-GAM)/GAM)-1.
      IF (FEMS1.LE.0.) GO TO 130
      EMS=EM1*SQRT(FEMS1/FEMS2)
      J=J+1
      IF (J.EQ.80) GO TO 290
      GO TO 140
290   CONTINUE
      GO TO 310
C
C     SOLUTION OBTAINED
C
C     DETERMINE B.L. PROPERTIES AT SEPARATION
300   CALL PRFL (UTUESS,PXS,EMS,PSP1(J),2,YY)
C
C     DETERMINE DOWNSTREAM B.L. INTEGRAL PROPERTIES
C
      CALL FLUX (101,YY,EMS)
C
C     DETERMINE DEL3/DEL1 AND SEPARATION PRESSURES
C
      PSP1F=PSP1(J)
      CPCV=(PSP1F*(1.+.7*EMI*EMI*CP1)-1.)/(.7*EMI*EMI)
C
C     RESULTS FROM CONTROL VOLUME APPROACH
C
      CPRT(1)=CPCV
C
C     RESULTS FROM GOLDSCHMEID
C
      CPRT(2)=CPSG
C
C     RESULTS FROM MODIFIED PAGE METHOD
C
      CPRT(3)=CPSP
310   CONTINUE
      RETURN
      END
      SUBROUTINE PRFL (UTUEST,PX,EME,PKP1,IOPT,YY)
C
C     SUBROUTINE TO CALCULATE DISTRIBUTIONS OF PROPERTIES
C
      DIMENSION YY(201)
```

```
      COMMON /BCB/ TTRAT(201),PTRAT(201),PTRNS(201),URAT(201),EM(201),WR
     1(201),PHIR(201),TWTTE,GAMMA,BLMN,BLMON,VWVE1,C,DSD,DDSD,BK,EME1,SH
     2FAC,SIG1,SIGMA1,SIGS1
C
      PI=3.1415927
      EXP2=GAMMA/(GAMMA-1.)
      EXP3=1./(GAMMA-1.)
      GAM1=(GAMMA-1.)/2.
      GAM2=GAMMA+1.
      GAM3=GAMMA-1.
      G1=EME*EME
      SIGMA=GAM1*G1/(1.+GAM1*G1)
      SIGG1=SQRT(SIGMA)
      SIGG2=1./SIGG1
      SIGG3=ATAN(SIGG1/SQRT(1-SIGG1*SIGG1))
      URAT(1)=0.
      TTRAT(1)=TWTTE
      EM(1)=0.
      YY(1)=0.
      PTRAT(1)=(1./(1.+GAM1*EME1*EME1))**EXP2*PKP1
      PTRNS(1)=PTRAT(1)
      DO 40 I=2,101
      AI=I-1
      YY(I)=AI/100.
      URAT(I)=SIGG2*SIN(SIGG3-SIGG3*PX*(1.+COS(PI*YY(I)))+(1./BK)*UTUEST
     1*SIGG3*ALOG(YY(I)))
      TTRAT(I)=TWTTE+(1.-TWTTE)*ABS(URAT(I))
      U2=URAT(I)*URAT(I)
      EM(I)=SQRT(U2/((1./G1+GAM1)*TTRAT(I)-GAM1*U2))
      IF (URAT(I).LE.0.) EM(I)=-1.*EM(I)
      IF (IOPT-1) 40,40,10
C
C     CALCULATION OF TOTAL PRESSURE DOWNSTREAM OF NORMAL SHOCK
C
10    PTRAT(I)=((1.+GAM1*EM(I)*EM(I))/(1.+GAM1*EME1*EME1))**EXP2*PKP1
      IF (EM(I)-1.) 20,20,30
20    PTRNS(I)=PTRAT(I)
      GO TO 40
30    PTRNS(I)=(GAM2*EM(I)*EM(I)/2./(1.+GAM1*EME1*EME1))**EXP2*(GAM2/(2.
     1*GAMMA*EM(I)*EM(I)-GAM3))**EXP3*PKP1
40    CONTINUE
      RETURN
      END
      SUBROUTINE FLUX (K,Y,EME)
C
C     SUBROUTINE TO CALCULATE MASS AND MOMENTUM FLUX OF B.L.
C     ALSO CALCULATES DISPLACEMENT AND MOMENTUM THICKNESSES
C
      DIMENSION Y(201), YY(201), BLMR(201), BLMOR(201)
      COMMON /BCB/ TTRAT(201),PTRAT(201),PTRNS(201),URAT(201),EM(201),WR
     1(201),PHIR(201),TWTTE,GAMMA,BLMN,BLMON,VWVE1,C,DSD,DDSD,BK,EME1,SH
     2FAC,SIG1,SIGMA1,SIGS1
C
      DO 10 I=1,K
      PRAT=1.
      TTOT=1.+(GAMMA-1.)*EM(I)*EM(I)/2.
      TTOTE=1.+(GAMMA-1.)*EME*EME/2.
      TOTE=TTOTE*TTRAT(I)/TTOT
      RHRAT=PRAT/TOTE
      BLMR(I)=RHRAT*URAT(I)
      BLMOR(I)=BLMR(I)*URAT(I)
      IF (URAT(I).LE.0.) BLMOR(I)=-BLMOR(I)
      YY(I)=Y(I)
10    CONTINUE
      DO 20 I=1,K
      CALL INTEG (I,YY,BLMR,AREA1)
```

```
      CALL INTEG (I,YY,BLMOR,AREA2)
      WR(I)=AREA1
      PHIR(I)=AREA2
20    CONTINUE
      BLMN=AREA1
      BLMON=AREA2
      DSD=1.-BLMN
      DDSD=BLMN-BLMON
      SHFAC=DSD/DDSD
      RETURN
      END
      SUBROUTINE INTEG (K,Y,Z,AREA)
C
C     INTEGRATION USING SIMPSON'S RULE
C
      DIMENSION Y(201), Z(201)
C
      IF (K.GE.5) GO TO 10
      IF (K.EQ.1) GO TO 80
      IF (K.EQ.2) GO TO 90
      IF (K.EQ.3) GO TO 100
      IF (K.EQ.4) GO TO 110
10    AK=K
      BK=AK/2.
      KK=BK
      CK=KK
      IF (BK-CK) 30,20,30
C
C     K IS EVEN
C
20    N=K-1
      GO TO 40
C
C     K IS ODD
C
30    N=K
40    ODD=0.
      EVEN=0.
      J=N-3
      DO 50 I=2,J,2
      EVEN=EVEN+Z(I)
      ODD=ODD+Z(I+1)
50    CONTINUE
      AREA=(Y(2)-Y(1))/3.*(Z(1)+Z(N)+4.*(EVEN+Z(N-1))+2.*ODD)
      IF (BK-CK) 70,60,70
C
C     K IS EVEN
C
60    AREA=AREA+(Y(K)-Y(K-1))*(Z(K)+Z(K-1))/2.
      RETURN
C
C     K IS ODD
C
70    RETURN
80    AREA=0.
      RETURN
90    AREA=(Y(2)-Y(1))*(Z(2)+Z(1))/2.
      RETURN
100   AREA=(Y(2)-Y(1))*(Z(3)+4.*Z(2)+Z(1))/3.
      RETURN
110   AREA=(Y(2)-Y(1))*((Z(4)+Z(3))/2.+(Z(3)+4.*Z(2)+Z(1))/3.)
      RETURN
      END
      SUBROUTINE SEP (NN,XA,RAD,CP,XSEP,AMIN,GAMMA,TTO,PT,RADO,DSTAR,Y,A
     1NA,IJET,NEXT,RI,UJ,C)
C
```

```
      DIMENSION DSTAR(1), RADO(1), XA(1), RAD(1), CP(1), Y(1), X834(201)
     1, Y834(201), RI(1), UJ(1)
      COMMON /SAVE/ SB(201),SC(201),YJB(201),XINS,XSEPSV(20),DELSV(20),Y
     1OUT(201)
C
      DO 10 I=1,NN
10    Y(I)=RAD(I)
      IF (XSEP.GT.0.) GO TO 30
      DO 20 I=1,NN
20    RADO(I)=Y(I)+DSTAR(I)
      GO TO 120
30    CONTINUE
      DO 40 I=1,NN
      IS=I
      IF (XSEP-XA(I)) 50,40,40
40    RADO(I)=Y(I)+DSTAR(I)
      GO TO 120
50    RTAN=(Y(IS-1)-Y(IS))/(XA(IS)-XA(IS-1))
      RTAN=ATAN(RTAN)
      YSEP=Y(IS)+(XSEP-XA(IS))*((Y(IS-1)-Y(IS))/(XA(IS-1)-XA(IS)))
      IC=1
      X834(1)=XSEP
      Y834(1)=YSEP
      DO 60 I=IS,NN
      IC=IC+1
      X834(IC)=XA(I)
60    Y834(IC)=Y(I)
      RADDEG=180./3.1415926
      RTAN=RTAN*RADDEG
      NEIN=NEXT-IS+2
      IJJB=0
      IF (ANA.GE.1.) IJJB=IJET
      IF (ANA.GE.9.) GO TO 70
      CALL B834 (IC,AMIN,GAMMA,TTO,PT,RTAN,X834,Y834,CP(IS-1),YOUT,IJJB,
     1NEIN,RI,UJ,C,ANA)
70    CONTINUE
      IF (ANA.EQ.1) GO TO 100
      JB=2
      DO 90 I=IS,NN
      Y(I)=YOUT(JB)
      RADO(I)=YOUT(JB)+DSTAR(I)
      IF (Y(I).LT.Y(I-1)) GO TO 80
      IF (RADO(I).GT.RADO(I-1)+Y(I)-Y(I-1)) RADO(I)=RADO(I-1)+Y(I)-Y(I-1
     1)
      GO TO 90
80    IF (RADO(I).GT.RADO(I-1)) RADO(I)=RADO(I-1)
90    JB=JB+1
      GO TO 120
100   CONTINUE
      JB=2
      API=2.*Y(IS)*DSTAR(IS)+DSTAR(IS)**2
      DO 110 I=IS,NN
      Y(I)=YOUT(JB)
      ARGF=4.0*Y(I)**2+4.0*API
      DSTAR(I)=(-2.*Y(I)+SQRT(ARGF))/2.0
      RADO(I)=YOUT(JB)+DSTAR(I)
      IF (RADO(I).GT.RADO(I-1)) RADO(I)=RADO(I-1)
110   JB=JB+1
120   CONTINUE
C     WRITE (6,150) ANA
C     WRITE (6,130) XSEP,YSEP
C     WRITE (6,140) (I,XA(I),RAD(I),Y(I),DSTAR(I),RADO(I),I=1,NN)
      RETURN
C
C
      END
```

```
      SUBROUTINE B834 (NST,FMS,GAMMA,TTO,PT,ABOD,XL,RAD,CPIN,YSTR,IJET,N
     1EXT,RI,UJ,C,ANA)
C
C     AXISYMMETRIC SEPARATION ANGLE PROGRAM
C
      COMMON /SAVE/ SB(201),SC(201),Y(201),XINS
      DIMENSION XSTR(201), YSTR(201), H1V(201), UEV(201), UBUEV(201), PS
     1IV(201), P1V(201), AME1V(201), XJET(201), CPIN(1), RI(1), UJ(1), X
     2L(201), RAD(201)
C
      NSEP=1
      NPT=101
      EPSLN=.00001
      DRJDX=-.05
      DUMDX=400.
      DELLOC=0.
      ILSV=0
      SLOPL=0.
      DO 10 I=1,20
10    H1V(I)=0.
      DEGRAD=3.1415926/180.
      DO 20 I=1,NST
      XSTR(I)=XL(I)
20    YSTR(I)=RAD(I)
      ABOD=ABOD*DEGRAD
      ABODSV=ABOD
      AT=SQRT(GAMMA*32.174*53.35*TTO)
      ISTOP=0
      PS=PT*(1.+(GAMMA-1.)*.5*FMS**2)**(GAMMA/(1.-GAMMA))
      IF (NSEP.LE.0) NSEP=1
      IL=NSEP-1
      IUE=0
      PSISV=0.
      DO 30 I=NSEP,NST
      P1V(I)=PS*(1.+GAMMA*.5*CPIN(I)*FMS**2)
      POWER=(1.-GAMMA)/GAMMA
      AME1V(I)=((P1V(I)/PT)**POWER-1.)*2./(GAMMA-1.)
      AME1V(I)=SQRT(AME1V(I))
30    UEV(I)=AME1V(I)*AT/SQRT(1.+.5*(GAMMA-1.)*AME1V(I)**2)
40    DELPSI=.05
C
C           ASSUME AN INITIAL SEPARATION ANGLE, PSIOLD; AND AN
C           INCREMENT , DELPSI
C
      IL=IL+1
      IUE=IUE+1
      P1=P1V(IL)
      P2=PS*(1.+GAMMA*.5*CPIN(IL+1)*FMS**2)
      POWER=(1.-GAMMA)/GAMMA
      AME1=AME1V(IL)
      AME2=((P2/PT)**POWER-1.)*2./(GAMMA-1.)
      AME2=SQRT(AME2)
      UE=UEV(IL)
      SIGMA=12.*(1.+.2298*AME1)
      IBAD=0
      I=0
      PSIOLD=.0
      IF (IL.GT.1) PSIOLD=PSIV(IL-1)-DELPSI
      PSIOLD=PSIOLD+DELPSI
50    CONTINUE
      I=I+1
C
C     CALCULATE H1
C
      IF (IL.GT.1.OR.I.GT.1) GO TO 60
      XSTR(IL)=XL(IL)
```

```
      YSTR(IL)=RAD(IL)
60    CONTINUE
      IF (IL-2) 70,80,90
70    H1=0.
      GO TO 90
80    ANGLE1=PSIV(1)
      H1=TAN(ANGLE1)*SQRT((XL(2)-XL(1))**2+(RAD(2)-RAD(1))**2)
90    CONTINUE
      RHO0=1.
      RHO1=1.
      RHO2=1.
      IF (I.GE.100) GO TO 260
      UBUE=0.
      ALPJ=(.2090+.0226*AME1+.308*UBUE)/SIGMA
      ALPHA=PSIOLD-ALPJ
      DELTA=PSIOLD/FLOAT(NPT-1)
      ICNT=0
100   ICNT=ICNT+1
      UBUEO=UBUE
      SUM1=0.
      SUM2=0.
      INTJB=NPT-1
C
C           USE SIMPSON*S RULE TO INTEGRATE THE CONTINUITY EQUATION
C           FOR UB/UE
C
      DO 120 J=2,INTJB,2
      THET0=DELTA*(J-2)
      THET1=DELTA*(J-1)
      THET2=DELTA*(J-0)
      ARG0=SIGMA*(THET0-ALPHA)
      ARG1=SIGMA*(THET1-ALPHA)
      ARG2=SIGMA*(THET2-ALPHA)
      XERF0=.5*(1.+ERT(ARG0))
      XERF1=.5*(1.+ERT(ARG1))
      XERF2=.5*(1.+ERT(ARG2))
      IF (ICNT.EQ.1) GO TO 110
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF0-UBUE)**2*(UE/AT)**2
      POWER=1./(GAMMA-1.)
      DEN=1.-(GAMMA-1.)*.5*(UE/AT)**2
      RHO0=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF1-UBUE)**2*(UE/AT)**2
      RHO1=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF2-UBUE)**2*(UE/AT)**2
      RHO2=(ANUM/DEN)**POWER
110   CONTINUE
      AX0=1.+H1*THET0/(RAD(IL)*PSIOLD)
      AX1=1.+H1*THET1/(RAD(IL)*PSIOLD)
      AX2=1.+H1*THET2/(RAD(IL)*PSIOLD)
      SUM1=SUM1+(DELTA/3.)*(RHO0*XERF0*AX0+4.*RHO1*XERF1*AX1+RHO2*XERF2*
     1AX2)
      SUM2=SUM2+(DELTA/3.)*(RHO0*(XERF0-1.)*AX0+4.*RHO1*(XERF1-1.)*AX1+R
     1HO2*(XERF2-1.)*AX2)
120   CONTINUE
      UBUE=-SUM1/SUM2
      IF (ICNT.GT.10) GO TO 130
      IF (ABS(UBUEO-UBUE).GT.ABS(.001*(UBUEO+UBUE)*.5)) GO TO 100
C
C     THETA ITERATION
C
130   THETA=0.
      IF (UBUE.GT.1.0) UBUE=1.0
      ICNT=0
      DELTH=DELTA
      RIGHT=2.*UBUE/(1.+UBUE)-1.
140   ARG=SIGMA*(THETA-ALPHA)
```

```
      ALEFT=ERT(ARG)
      IF (ICNT.GE.100) GO TO 170
      IF (ABS(ALEFT).GT.ABS(RIGHT)) GO TO 150
      IF (ABS(RIGHT-ALEFT).LE.ABS(.01*(RIGHT+ALEFT)*.5)) GO TO 160
      THETA=THETA-DELTH
      DELTH=DELTH/10.
150   THETA=THETA+DELTH
      ICNT=ICNT+1
      IF (THETA.GE.PSIOLD) GO TO 160
      GO TO 140
160   CONTINUE
      ICNT=THETA/PSIOLD*100
      IF (ICNT/2*2.NE.ICNT) ICNT=ICNT+1
170   CONTINUE
      SUM3=0.
C
C           USE SIMPSON*S RULE TO INTEGRATE THE MOMENTUM EQUATION
C
      DO 180 J=2,ICNT,2
      THET0=DELTA*(J-2)
      THET1=DELTA*(J-1)
      THET2=DELTA*(J-0)
      ARG0=SIGMA*(THET0-ALPHA)
      ARG1=SIGMA*(THET1-ALPHA)
      ARG2=SIGMA*(THET2-ALPHA)
      XERF0=.5*(1.+ERT(ARG0))
      XERF1=.5*(1.+ERT(ARG1))
      XERF2=.5*(1.+ERT(ARG2))
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF0-UBUE)**2*(UE/AT)**2
      POWER=1./(GAMMA-1.)
      DEN=1.-(GAMMA-1.)*.5*(UE/AT)**2
      RHO0=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF1-UBUE)**2*(UE/AT)**2
      RHO1=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF2-UBUE)**2*(UE/AT)**2
      RHO2=(ANUM/DEN)**POWER
      AX0=1.+(H1*THET0)/(RAD(IL)*PSIOLD)
      AX1=1.+(H1*THET1)/(RAD(IL)*PSIOLD)
      AX2=1.+(H1*THET2)/(RAD(IL)*PSIOLD)
      TEMP=(DELTA/3.)*(RHO0*COS(THET0)*((1.+UBUE)*XERF0-UBUE)**2*AX0+4.*
     1RHO1*COS(THET1)*((1.+UBUE)*XERF1-UBUE)**2*AX1+RHO2*COS(THET2)*((1.
     2+UBUE)*XERF2-UBUE)**2*AX2)
      SUM3=SUM3+TEMP
180   CONTINUE
      SUM4=0.
      IF (ICNT.GE.100) GO TO 200
      ICNT=ICNT+2
      DO 190 J=ICNT,INTJB,2
      THET0=DELTA*(J-2)
      THET1=DELTA*(J-1)
      THET2=DELTA*(J-0)
      ARG0=SIGMA*(THET0-ALPHA)
      ARG1=SIGMA*(THET1-ALPHA)
      ARG2=SIGMA*(THET2-ALPHA)
      XERF0=.5*(1.+ERT(ARG0))
      XERF1=.5*(1.+ERT(ARG1))
      XERF2=.5*(1.+ERT(ARG2))
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF0-UBUE)**2*(UE/AT)**2
      POWER=1./(GAMMA-1.)
      DEN=1.-(GAMMA-1.)*.5*(UE/AT)**2
      RHO0=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF1-UBUE)**2*(UE/AT)**2
      RHO1=(ANUM/DEN)**POWER
      ANUM=1.-(GAMMA-1.)*.5*((1.+UBUE)*XERF2-UBUE)**2*(UE/AT)**2
      RHO2=(ANUM/DEN)**POWER
      AX0=1.+(H1*THET0)/(RAD(IL)*PSIOLD)
```

```
      AX1=1.+(H1*THET1)/(RAD(IL)*PSIOLD)
      AX2=1.+(H1*THET2)/(RAD(IL)*PSIOLD)
      TEMP=(DELTA/3.)*(RHO0*COS(THET0)*((1.+UBUE)*XERF0-UBUE)**2*AX0+4.*
     1RHO1*COS(THET1)*((1.+UBUE)*XERF1-UBUE)**2*AX1+RHO2*COS(THET2)*((1.
     2+UBUE)*XERF2-UBUE)**2*AX2)
      SUM4=SUM4+TEMP
190   CONTINUE
200   CONTINUE
      SUM5=SUM3+SUM4
      ANUM=1.-(GAMMA-1.)*.5*(1.+UBUE)**2*(UE/AT)**2
      DEN=1.-(GAMMA-1.)*.5*(UE/AT)**2
      RHOT=.0005821*PT/TTO
      RHORT=(1.+(GAMMA-1.)*.5*AME1**2)**(1./(1.-GAMMA))
      RHOE=RHORT*RHOT
      ANUM=1.-(GAMMA-1.)*.5*UBUE**2*(UE/AT)**2
      RHOBT=(ANUM/DEN)**(1./(GAMMA-1.))
      AMACB=RHOBT*UBUE**2
C     CF
      IF (IJET.EQ.0) GO TO 210
      IF (IL.LT.NEXT) GO TO 210
      CF=0.
      GO TO 220
210   CONTINUE
      ENU=.56/3600.
      XX=1.
      REX=UBUE*UE/(ENU*XX)
      CF=1.328/SQRT(REX)
220   SKIN=CF*.5*AMACB
      DPDX=(P2-P1)/(XL(IL+1)-XL(IL))
      H2=H1
      HA=H1
      DIST=SQRT((XL(IL+1)-XL(IL))**2+(RAD(IL+1)-RAD(IL))**2)
      IF (IL.EQ.1) GO TO 230
      H2=H1*TAN(PSIOLD)/TAN(PSIV(IL-1))+DIST*TAN(PSIOLD)
C
C     AB  PERPENDICULAR DISTANCE FROM SEPARATION SLOPE LINE TO CONTOUR
C     DB DISTANCE ALONG SLOPE LINE AT SEPARATION
C
      DB=ABS(H2-H1)/TAN(PSIOLD)
      SLOPL=(RAD(IL)-RAD(IL+1))/(XL(IL+1)-XL(IL))
      SLOPL=ATAN(SLOPL)
      AMUS=SLOPL-ABODSV
      AB=DB*TAN(AMUS)
      HA=H1
230   PRESS=DPDX/(RHOE*UE**2)*(HA+HA**2/(2.*RAD(IL)))*COS(SLOPL)
      IF (PRESS.LT.0.) PRESS=0.
      SHEAR=(1.+HA/RAD(IL))*(.1481-.0478*AME1+.1278*UBUE-.1632*AME1*UBUE
     1+.399*UBUE**2-.0239*EXP(-5.0*AME1))/SIGMA
      SHEAR=SHEAR*COS(PSIOLD)
      SHRJT=0.
      OLDANS=SHEAR-PRESS-SKIN+SHRJT
C
C           COMPARE THE RIGHT AND LEFT SIDE OF MOMENTUM EQUATION
C
      AMULT=P2*AME2**2/(P1*AME1**2)
      SUM5NW=SUM5*AMULT
      IF (IL.EQ.1) GO TO 240
      X01=H1/PSIV(IL-1)
      X02=H2/PSIOLD
      RATIO=(1.+H1/RAD(IL))/(1.+H1V(IL-1)/RAD(IL-1))
      SUM5NW=(X02*AMULT*SUM5-X01*RATIO*SUMSV)/(X02-X01)
240   CONTINUE
      GLI=OLDANS-SUM5NW
      IF (IBAD.EQ.0) GO TO 250
      IJB=1
      GO TO 300
```

```
250   CALL NEWRAP (I,PSIOLD,GLI,EPSLN,PSINEW,IJB)
      IF (PSINEW.LT.0.) GO TO 260
      IF (PSINEW.GT.ABODSV) GO TO 260
      IF (I.GE.100) GO TO 260
      GO TO 300
C
C     ITERATION FAILED, USE ANGLE FROM PREVIOUS ITERATION
C
260   CONTINUE
      IF (AME1.LT..2) GO TO 280
      IF (AME1.LT..5) GO TO 270
      SLOPE=(8.4-15.625)/(1.-.5)
      PSINEW=15.625+SLOPE*(AME1-.5)
      GO TO 290
270   SLOPE=(15.625-17.5)/(.5-.2)
      PSINEW=17.5+SLOPE*(AME1-.2)
      GO TO 290
280   PSINEW=17.5
290   WRITE (6,370) PSINEW
      PSINEW=PSINEW*DEGRAD
      IF (PSINEW.GT.ABODSV) PSINEW=ABODSV
      PSIOLD=PSINEW
      IBAD=1
      IF (ANA.GE.8.) WRITE (6,380)
      GO TO 50
300   PSIOLD=PSINEW
      IF (IJB.EQ.0) GO TO 50
      DELOLD=DELLOC
      IF (IL.GT.1) DELLOC=ABS(H2-H1)/DB
      DELLOC=ATAN(DELLOC)
      IF (IL.EQ.1) DELLOC=PSIOLD
      IF (IL.EQ.1) DELOLD=0.
      ASTR=ABOD-(DELLOC-DELOLD)
C
C     CALCULATE DISCRIMINATING STREAMLINE
C
      XSTR(IL+1)=XL(IL+1)
      A=SQRT((XL(IL+1)-XL(IL))**2)
      B=TAN(ASTR)*A
      YSTR(IL+1)=YSTR(IL)-B
      H1P=YSTR(IL)-RAD(IL)
      XIN=XSTR(NST)*1.001
      IX=NST
      IF (YSTR(IL+1).GT.RAD(IL+1)) GO TO 320
      H2P=ABS(YSTR(IL+1)-RAD(IL+1))
      AX=(XL(IL+1)-XL(IL))/(1.+H2P/H1P)
      XSTR(IL+1)=XSTR(IL)+AX
      XIN=XSTR(IL+1)
      XINS=XIN
      CX=(RAD(IL)-RAD(IL+1))/(1.+H2P/H1P)
      YSTR(IL+1)=RAD(IL)-CX
      ISTOP=1
      IX=IL+1
      IF (IX.GT.NST) GO TO 320
      DO 310 JB=IX,NST
      XSTR(JB+1)=XL(JB)
310   YSTR(JB+1)=RAD(JB)
320   CONTINUE
      SUMSV=SUMS
      UBUEV(IL)=UBUE
      H1V(IL)=H1
      PSIV(IL)=PSIOLD
      ABOD=ASTR
      PSISV=PSIOLD
      IF (ISTOP.EQ.1) GO TO 360
      IF (IJET.EQ.0) GO TO 340
```

```
      IF (IL+1.LT.NEXT) GO TO 340
C
C     CALL JET ENTRAINMENT IF OPTION TURNED ON
C
      RJ=RAD(IL+1)
      RD=YSTR(IL+1)
      NSIN=NST-NEXT+1
      DO 330 I=NEXT,NST
330   XJET(I)=XL(I)-XL(NEXT)
      UM=-UBUEV(IL)*UE
      DRDDX0=(YSTR(NEXT)-YSTR(NEXT-1))/(XL(NEXT)-XL(NEXT-1))
      CALL JET (RJ,UM,C,DRJDX,DUMDX,RD,NSIN,XJET(NEXT),UEV(NEXT),YSTR(NE
     1XT),RI,UJ,DRDDX0)
      GO TO 360
340   CONTINUE
C
C     DETERMINE IF ITERATION COMPLETE
C
      IF (IL.EQ.1) GO TO 350
      DB=ABS(H2-H1)/TAN(PSIOLD)
      AMUS=SLOPL-ABODSV
      AB=DB*TAN(AMUS)
      H1=H2+AB
      IF (H1.GT.0.) GO TO 350
      IF (ILSV.EQ.0) ILSV=IL
      H1=H1V(ILSV)*.01
350   IF (IL+1.LT.NST) GO TO 40
360   RETURN
C
C
370   FORMAT (1H ,6HPSINEW,F12.4)
380   FORMAT (1H0,46HITERATION FOR DISCRIMINATING STREAMLINE ANGLE ,28HF
     1AILED, USING DEFAULT VALUE.,/17H  TRY DECREASING ,36HSTEP SIZE (MO
     2RE POINTS ON AFTERBODY))
      END
      FUNCTION ERT (X)
C
C        THIS FUNCTION ROUTINE OBTAINS VALUE OF THE ERROR FUNCTION
C        WITH ARGUMENT X USING LIBRARY SUBROUTINE ERF
C
      CALL ERF (X,Y)
      ERT=Y
      RETURN
      END
      SUBROUTINE NEWRAP (ICNT,X,FUNC,TOLL,XZERO,IE)
C
      IE=0
      IF (ICNT.GT.100) STOP
      IF (ICNT-2) 10,20,30
10    FUN1=FUNC
      X1=X
      IF (X.EQ.0.) X=100.*TOLL
      XZERO=X+.1*X
      GO TO 90
20    CONTINUE
      FUN2=FUNC
      X2=X
      GO TO 80
30    CONTINUE
      IF (FUN1*FUN2) 50,40,40
40    FUN1=FUN2
      FUN2=FUNC
      X1=X2
      X2=X
      GO TO 80
50    CONTINUE
```

```
      IF (FUNC*FUN2) 70,60,60
60    FUN2=FUNC
      X2=X
      GO TO 80
70    FUN1=FUNC
      X1=X
C
C     CALCULATE DERIVATIVE
C
80    DERV=(FUN2-FUN1)/(X2-X1)
      XZERO=X2-FUN2/DERV
      IF (ABS(XZERO-X).LT.ABS((XZERO+X2)*.5)*TOLL) IE=1
90    RETURN
      END
      SUBROUTINE JET (RJ,UM,C,DRJDX,DUMDX,RD,NBIN,XIN,UE,YSTR,RJA,UJA,DR
     1DDX0)
C
      DIMENSION XIN(201), YSTR(201), RJA(201), UJA(201), UE(201)
      REAL L0,L1,L2
C
      TOL=.001
      DRDDX=.1
      ICNT=0
      ISTA=1
      RDSV=RD
      X=XIN(ISTA)
      DL2DX=.25
      DRCDX=DRJDX+DL2DX
      DREDX=0.
      UE0=UE(1)
      R0=RJ
      A=(.5714*RD+.2286*R0)/(2.*(.2*RD+.05*R0))
      B=(RD+R0)*UM/(2.*(.2*RD+.05*R0)*(UE0-UM))
      XI=A-SQRT(A**2+B)
      ETA=XI**(1./1.5)
      ETA2=(1.-SQRT(UE0/(UE0-UM)))**(1./1.5)
      IF (ETA.LT.ETA2) ETA=ETA2
      L0=(RD-R0)/ETA
      RE=R0+L0
10    CONTINUE
      UE1=UE(ISTA)
      UJ=UJA(ISTA)
      DUEDX=(UE(ISTA+1)-UE(ISTA))/(XIN(ISTA+1)-XIN(ISTA))
      DUJDX=(UJA(ISTA+1)-UJA(ISTA))/(XIN(ISTA+1)-XIN(ISTA))
      ISTA=ISTA+1
      L2=.25*X
      RC=RJ+L2
      DELUE=UE1-UM
      L1=RE-RJ-L2
C
C     DUMDX  LOOP
C
      ICNT=0
20    CONTINUE
      ICNT=ICNT+1
      DL1DX=C
      DDUEDX=DUEDX-DUMDX
      DLEUJ=UJ-UM
      DDUJXX=DUJDX-DUMDX
      AJB=L2*(UM+.55*DLEUJ)+UJ*RJ
      BJB=(RJ*(UM+.55*DLEUJ)+.3572*DLEUJ*L2)*DL2DX+(RC*L2-.5*L2**2)*DUMD
     1X+(.55*RC*L2-.3714*L2**2)*DDUJXX
      DRJDX=(.043*UJ*RC-BJB)/AJB
      IF (ISTA.GT.2) GO TO 30
      DRDDX=DRDDX0
      TRM1=(RC*(UM+DELUE*(2.*ETA**1.5-ETA**3))+L1*ETA*(UM+DELUE*(2.*ETA*
```

```
     1*1.5-ETA**3)))*(DRDDX-DL2DX-ETA*DL1DX)+(RC*(UM*ETA+DELUE*(2./2.5*E
     2TA**2.5-1./4.*ETA**4))+2.*L1*(.5*UM*ETA**2+DELUE*(2./3.5*ETA**3.5-
     31./5.*ETA**5)))*DL1DX
      TRM2=L1*(UM*ETA+DELUE*(2./2.5*ETA**2.5-1./4.*ETA**4))*DL2DX+(RC*L1
     1*ETA+.5*L1**2*ETA**2)*DUMDX+(RC*L1*(2./2.5*ETA**2.5-1./4.*ETA**4)+
     2L1**2*(2./3.5*ETA**3.5-1./5.*ETA**5))*DDUEDX
      B=TRM1+TRM2
      A=L1*(UM*ETA+DELUE*(2./2.5*ETA**2.5-1./4.*ETA**4))-RC*(UM+DELUE*(2
     1.*ETA**1.5-ETA**3))-L1*ETA*(UM+DELUE*(2.*ETA**1.5-ETA**3))
      XZ=(BJB-(AJB*(BJB+B))/(AJB+A))/(UJ*RC)
   30 CONTINUE
      DRJDX=(UJ*XZ*RC-BJB)/AJB
      DRCDX=DRJDX+DL2DX
      DDUEDX=DUEDX-DUMDX
      TRM1=(RC*(UM+.55*DLEUJ)-2.*L2*(.5*UM+.3714*DLEUJ))*DL2DX
      TRM2=L2*(UM+.55*DLEUJ)*DRCDX+(RC*L2-.5*L2**2)*DUMDX
      TRM3=(.55*RC*L2-.3714*L2**2)*DDUJXX
      UDR1X=TRM1+TRM2+TRM3
      DREDX=DRJDX+C+DL1DX
      IF (ISTA.GT.2) GO TO 40
      B1=L1*(UM+DELUE*(2./2.5*ETA**1.5-1./4.*ETA**3))*ETA*DRCDX+(RC*L1*E
     1TA+.5*L1**2*ETA**2)*DUMDX+(RC*L1*(2./2.5*ETA**2.5-1./4.*ETA**4)+L1
     2**2*(2./3.5*ETA**3.5-1./5.*ETA**5))*DDUEDX
      ZZZ=RC*L1*(UM+DELUE*(2.*ETA**1.5-ETA**3))+L1**2*ETA*(UM+DELUE*(2.*
     1ETA**1.5-ETA**3))
      Q1=-(UJ*RJ*DRJDX+UDR1X+B1)/ZZZ
      P1=-(RC*(UM*ETA+DELUE*(2./2.5*ETA**2.5-1./4.*ETA**4))+2.*L1*(.5*UM
     1*ETA**2+DELUE*(2./3.5*ETA**3.5-1./5.*ETA**5)))/ZZZ
      DL1DX=(DRDDX0-DRJDX-DL2DX-L1*Q1)/(ETA+P1*L1)
   40 DREDX=DRJDX+C+DL1DX
      DL1DX=C
      DREDX=DRJDX+C+DL1DX
      IF (DRDDX.EQ.0.) ETA=(1.-SQRT(UE1/DELUE))**(1./1.5)
      TRM1=(RC*(UM*ETA+DELUE*(2./2.5*ETA**2.5-ETA**4/4.))+2.*L1*(.5*UM*E
     1TA**2+DELUE*(2./3.5*ETA**3.5-ETA**5/5.)))*DL1DX
      TRM2=L1*(UM+DELUE*(2./2.5*ETA**1.5-ETA**3/4.))*ETA*DRCDX+(RC*L1*ET
     1A+.5*L1**2*ETA**2)*DUMDX
      TRM3=(RC*L1*(2./2.5*ETA**2.5-ETA**4/4.)+L1**2*(2./3.5*ETA**3.5-ETA
     1**5/5.))*DDUEDX
      WWW=TRM1+TRM2+TRM3
      ZZZ=RC*L1*(UM+DELUE*(2.*ETA**1.5-ETA**3))+L1**2*ETA*(UM+DELUE*(2.*
     1ETA**1.5-ETA**3))
      DLEUJ=UJ-UM
      DDUJXX=DUJDX-DUMDX
      TRM1=(RC*(UM+.55*DLEUJ)-2.*L2*(.5*UM+.3714*DLEUJ))*DL2DX
      TRM2=L2*(UM+.55*DLEUJ)*DRCDX+(RC*L2-.5*L2**2)*DUMDX
      TRM3=(.55*RC*L2-.3714*L2**2)*DDUJXX
      UDR1X=TRM1+TRM2+TRM3
      UDR2X=(2.*L1*(.5*UM+.3714*DELUE)+RC*(UM+.55*DELUE))*DL1DX+L1*(UM+.
     155*DELUE)*DRCDX+(.5*L1**2+RC*L1)*DUMDX+(.3714*L1**2+.55*RC*L1)*DDUJ
     2EDX
      IF (DRDDX.EQ.0.) GO TO 50
      DEDX=-(XZ*UJ*RC+WWW)/ZZZ
      DRDDX=ETA*DREDX+(1.-ETA)*DRCDX+L1*DEDX
   50 CONTINUE
      DLEUJ=UJ-UM
      DDUJXX=DUJDX-DUMDX
      TRM1=L2*(UM**2+1.1*UM*DLEUJ+.4156*DLEUJ**2)*DRCDX
      TRM2=(RC*(UM**2+1.1*UM*DLEUJ+.4156*DLEUJ**2)-2.*L2*(.5*UM**2+.7428
     1*UM*DLEUJ+.3096*DLEUJ**2))*DL2DX
      TRM3=(RC*L2*(2.*UM+1.1*DLEUJ)-L2**2*(UM+.7428*DLEUJ))*DUMDX
      TRM4=(RC*L2*(1.1*UM+.8312*DLEUJ)-L2**2*(.7428*UM+.6192*DLEUJ))*DDUJ
     1JXX
      U2DR1X=TRM1+TRM2+TRM3+TRM4
      TRM1=((UM**2+1.4856*UM*DELUE+.6192*DELUE**2)*L1+(UM**2+1.1*UM*DELU
     1E+.4156*DELUE**2)*RC)*DL1DX
```

```
       TRM2=L1*(UM**2+1.1*UM*DELUE+.4156*DELUE**2)*DRCDX+L1**2*(UM+.7428*
      1DELUE)*DUMDX
       TRM3=L1*RC*(2.*UM+1.1*DELUE)*DUMDX+L1**2*(.6192*DELUE+.7428*UM)*DD
      1UEDX+RC*L1*(.8312*DELUE+1.1*UM)*DDUEDX
       U2DR2X=TRM1+TRM2+TRM3
       AMRHO=UJ*RJ*DRJDX+(UDR1X+UDR2X)
       XMOMNT=UJ**2*RJ*DRJDX+(U2DR1X+U2DR2X)-RE**2*.5*UE1*DUEDX-UE1*AMRHO
       CALL NEWRAP (ICNT,DUMDX,XMOMNT,TOL,DUMDXN,IE)
       IF (ICNT.LE.30) GO TO 60
       ISTA=ISTA-1
       GO TO 130
60     CONTINUE
       IF (IE.NE.0) GO TO 70
       DUMDX=DUMDXN
       GO TO 20
70     CONTINUE
C
C      SOLVE  FOR  THE  RADIUS  OF  THE  DISCRIMINATING  STREAMLINE
C
       X=XIN(ISTA)
       DX=XIN(ISTA)-XIN(ISTA-1)
       L2=C*X
       IF (ETA.LE.ETA2) GO TO 80
       IF (DRDDX.EQ.0.) GO TO 80
       RD=DRDDX*DX+RD
       ETA=DEDX*DX+ETA
       DELUE=UE(ISTA)-(DUMDX*DX+UM)
       ETA2=(1.-SQRT(UE(ISTA)/DELUE))**(1./1.5)
       IF (ETA.LT.ETA2) GO TO 80
       GO TO 90
80     CONTINUE
       DRDDX=0.
90     UM=DUMDX*DX+UM
       IF (UM.GE.0.) GO TO 170
       RJ=RJ+DRJDX*DX
       DREDX=DRJDX+2.*C
       RE=RE+DREDX*DX
       IF (DRDDX.NE.0.) GO TO 100
       L1=RE-RJ-L2
       IF (L1.LE.0.) GO TO 170
       DELUE=UE(ISTA)-UM
       ETA=(1.-SQRT(UE(ISTA)/DELUE))**(1./1.5)
       RD=RJ+L2+ETA*L1
       IF (RD.GT.RDSV) GO TO 170
100    CONTINUE
       YSTR(ISTA)=RD+RJA(ISTA)-R0-L2
       RDSV=RD
       ICNT=0
       IF (RD.LE.R0) GO TO 110
       IF (YSTR(ISTA).LE.RJA(ISTA)) GO TO 110
       GO TO 190
110    ISRT=ISTA
       DO 120 I=ISRT,NSIN
120    YSTR(I)=RJA(I)
       GO TO 200
130    SLOPE=(YSTR(ISTA-1)-YSTR(ISTA-2))/DX
       IF (SLOPE.GT.0.) GO TO 150
       ISRT=ISTA
       DO 140 I=ISRT,NSIN
       YSTR(I)=SLOPE*(XIN(I)-XIN(I-1))+YSTR(I-1)
140    IF (YSTR(I).LT.RJA(I)) YSTR(I)=RJA(I)
       GO TO 200
150    DO 160 I=ISTA,NSIN
160    YSTR(I)=YSTR(I-1)
       GO TO 200
170    DO 180 I=ISTA,NSIN
```

```
      L2=C*XIN(I)
      YSTR(I)=RDSV+RJA(I)-R0-L2
180   IF (YSTR(I).LT.RJA(I)) YSTR(I)=RJA(I)
      GO TO 200
190   CONTINUE
      IF (ISTA.LT.NSIN) GO TO 10
200   RETURN
      END
      SUBROUTINE SMINT (XA,YA,NA,NMIN,NMAX)
C
C     INTERFACE ROUTINE FOR VISCOUS PACKAGE AND SMOOTHING ROUTINES
C
      COMMON /SAVE/ SB(201),SC(201),YJB(201),XIN
      DIMENSION X1(201), X2(201), Y1(201), Y2(201), XA(1), YA(1), S(201)
     1, S1(201), Y22(201), Y22S(201), Y(201), Z(201), DDY(201), DY(201),
     2 DZ(201), DDZ(201), Z1(201)
C
      NA1=NA-1
      DO 10 I=1,NA1
      X1(I)=XA(I)
      X2(I)=XA(I+1)
      Y1(I)=YA(I)
10    Y2(I)=YA(I+1)
      K9=1
      NSMTH1=NMIN-4
      IF (NSMTH1.LT.2) NSMTH1=2
      NSMTH2=NA-1
      DO 20 I=NMAX,NA
      IF (XIN.LT.XA(I)) GO TO 30
      NSMTH2=I+6
20    CONTINUE
30    IF (NSMTH2.GT.NA-1) NSMTH2=NA-1
      IVSM=0
      K11=10
      CALL SMOOTH (X1,X2,Y1,Y2,K9,K11,NSMTH1,NSMTH2,IVSM,NA,S,S1,Y22,Y22
     1S,Y,Z,DDY,DY,DZ,DDZ,Z1)
      DO 40 I=1,NA1
      XA(I)=X1(I)
40    YA(I)=Y1(I)
      RETURN
      END
      SUBROUTINE SMOOTH (X1,X2,Y1,Y2,K9,K11,NSMTH1,NSMTH2,IVSM,NA,S,S1,Y
     122,Y22S,Y,Z,DDY,DY,DZ,DDZ,Z1)
C
      DIMENSION X1(NA), Y1(NA), Y2(NA), S(NA), S1(NA), X2(NA), Y22(NA),
     1Y22S(NA), Y(NA), Z(NA), NSMTH1(5), NSMTH2(5), DDY(NA), DY(NA), DZ(
     2NA), DDZ(NA), Z1(NA)
C
      DO 170 I=1,K9
      N5=NSMTH1(I)
      N6=NSMTH2(I)
      I1=MIN0(N5,N6)
      I2=MAX0(N5,N6)
      S(I1-1)=0.0
      IF (IVSM.EQ.1) GO TO 20
      DO 10 J=I1,I2
10    S(J)=S(J-1)+SQRT((X2(J)-X1(J))**2+(Y2(J)-Y1(J))**2)
      GO TO 40
20    DO 30 J=I1,I2
30    S(J)=S(J-1)+SQRT((X1(J)-X1(J-1))**2+(X2(J)-X2(J-1))**2)
40    CONTINUE
      II=I2-I1+1
      DELS=S(I2)/II
      I3=I2-1
      S1(I1-1)=0.0
      DO 50 J=I1,I3
```

```
50    S1(J)=S1(J-1)+DELS
      S1(I2)=S(I2)
      DO 90 J=I1,I3
      DO 60 K=I1,I3
      IF (S(K)-S1(J)) 60,70,70
60    CONTINUE
70    IF (IVSM.EQ.1) GO TO 80
      Y22(J)=Y1(K)+(Y2(K)-Y1(K))*(S1(J)-S(K-1))/(S(K)-S(K-1))
      GO TO 90
80    Y22(J)=Y1(K-1)+(Y1(K)-Y1(K-1))*(S1(J)-S(K-1))/(S(K)-S(K-1))
90    CONTINUE
      Y22(I2)=Y2(I2)
      IF (IVSM.EQ.1) Y22(I2)=Y1(I2)
      DO 100 J=1,II
100   Y(J)=Y22(J+I1-1)
      CALL MSMTH (Y,Z,II,K11,NA,DDY,DY,DZ,DDZ,Z1)
      DO 110 J=I1,I2
110   Y22S(J)=Z(J-I1+1)
      Y22S(I1-1)=Y2(I1-1)
      IF (IVSM.EQ.1) Y22S(I1-1)=Y1(I1-1)
      DO 160 J=I1,I3
      DO 120 K=I1,I3
      IF (S1(K)-S(J)) 120,130,130
120   CONTINUE
130   IF (IVSM.EQ.1) GO TO 140
      Y2(J)=Y22S(K-1)+(Y22S(K)-Y22S(K-1))*(S(J)-S1(K-1))/(S1(K)-S1(K-1))
      GO TO 150
140   CONTINUE
      Y1(J)=Y22S(K-1)+(Y22S(K)-Y22S(K-1))*(S(J)-S1(K-1))/(S1(K)-S1(K-1))
150   CONTINUE
      IF (IVSM.EQ.0) Y1(J)=Y2(J-1)
160   CONTINUE
170   CONTINUE
      RETURN
      END
      SUBROUTINE MSMTH (Y,Z,N,K,NA,DDY,DY,DZ,DDZ,Z1)
C
      DIMENSION Y(NA), Z(NA), DY(NA), DZ(NA), DDY(NA), DDZ(NA), Z1(NA)
C
      IF (N-5) 40,10,10
10    CALL RSMTH (Y,N,K,Z,NA,Z1)
      NM1=N-1
      NM2=N-2
      DO 20 I=1,NM1
      DY(I)=Y(I+1)-Y(I)
      DZ(I)=Z(I+1)-Z(I)
20    CONTINUE
      DO 30 I=1,NM2
      DDY(I)=DY(I+1)-DY(I)
      DDZ(I)=DZ(I+1)-DZ(I)
30    CONTINUE
40    RETURN
      END
      SUBROUTINE RSMTH (Y,N,K,Z,NA,Z1)
C
      DIMENSION Y(NA), Z(NA), Z1(NA)
C
      J=0
      DO 10 I=1,N
      Z1(I)=Y(I)
10    CONTINUE
      Z(1)=Y(1)
      Z(N)=Y(N)
20    J=J+1
      CALL SMTH (Z1,N,U,Z,NA)
      IF (U) 60,60,30
```

```
30    IF (J-K) 40,60,60
40    DO 50 I=1,N
      Z1(I)=Z(I)
50    CONTINUE
      GO TO 20
60    RETURN
      END
      SUBROUTINE SMTH (Y,N,U,Z,NA)
C
      DIMENSION Y(NA), Z(NA)
C
      F(ETA)=(1.0-ETA*ETA)**2
      G(ET)=0.5*ET*(1.0+ET)
C
      U=0.0
      J=3
10    A=ABS(Y(J-2)-2.0*Y(J-1)+2.0*Y(J+1)-Y(J+2))
      D=-Y(J-2)-4.0*Y(J-1)+10.0*Y(J)-4.0*Y(J+1)-Y(J+2)
      IF (A-ABS(D)) 20,40,40
20    FS=F(A/D)
      A=ABS(-Y(J-2)+Y(J-1)+Y(J+1)-Y(J+2))
      B=ABS(3.0*(Y(J-1)-2.0*Y(J)+Y(J+1)))
      IF (A-B) 30,40,40
30    GS=G(1.0-A/B)
      Z(J)=Y(J)-0.1*FS*GS*D
      U=1.0
      GO TO 50
40    Z(J)=Y(J)
50    J=J+1
      IF (J-(N-2)) 10,10,60
60    A=0.4*ABS(Y(4)-Z(4))
      D=0.2*(-6.0*Y(1)+10.0*Y(2)-2.0*Y(3)-Y(4)-Z(4))
      IF (A-ABS(D)) 70,80,80
70    FS=F(A/D)
      GS=G(A/ABS(D))
      Z(2)=Y(2)-0.5*FS*GS*D
      GO TO 90
80    Z(2)=Y(2)
90    A=.4*ABS(Y(N-3)-Z(N-3))
      D=.2*(-Y(N-3)-Z(N-3)-2.*Y(N-2)+10.0*Y(N-1)-6.*Y(N))
      IF (A-ABS(D)) 100,110,110
100   FS=F(A/D)
      GS=G(A/ABS(D))
      Z(N-1)=Y(N-1)-0.5*FS*GS*D
      GO TO 120
110   Z(N-1)=Y(N-1)
120   RETURN
      END
      SUBROUTINE IUNI(NMAX,N,X,NTAB,Y,IORDER,X0,Y0,IPT,IERR)
C*************************************************************************
C*                                                                       *
C*    PURPOSE:                                                           *
C*               SUBROUTINE IUNI USES FIRST OR SECOND ORDER              *
C*               LAGRANGIAN INTERPOLATION TO ESTIMATE THE VALUES         *
C*               OF A SET OF FUNCTIONS AT A POINT X0.  IUNI              *
C*               USES ONE INDEPENDENT VARIABLE TABLE AND A DEPENDENT     *
C*               VARIABLE TABLE FOR EACH FUNCTION TO BE EVALUATED.       *
C*               THE ROUTINE ACCEPTS THE INDEPENDENT VARIABLES SPACED    *
C*               AT EQUAL OR UNEQUAL INTERVALS.  EACH DEPENDENT          *
C*               VARIABLE TABLE MUST CONTAIN FUNCTION VALUES CORRES-     *
C*               PONDING TO EACH X(I) IN THE INDEPENDENT VARIABLE        *
C*               TABLE.  THE ESTIMATED VALUES ARE RETURNED IN THE Y0     *
C*               ARRAY WITH THE N-TH VALUE OF THE ARRAY HOLDING THE      *
C*               VALUE OF THE N-TH FUNCTION VALUE EVALUATED AT X0.       *
C*                                                                       *
C*       USE:                                                            *
```

```
C*              CALL IUNI(NMAX,N,X,NTAB,Y,IORDER,X0,Y0,IPT,IERR)
C*
C* PARAMETERS:
C*
C*       NMAX   THE MAXIMUM NUMBER OF POINTS IN THE INDEPENDENT
C*              VARIABLE ARRAY.
C*
C*          N   THE ACTUAL NUMBER OF POINTS IN THE INDEPENDENT
C*              ARRAY,WHERE N .LE. NMAX.
C*
C*          X   A ONE-DIMENSIONAL ARRAY, DIMENSIONED (NMAX) IN THE
C*              CALLING PROGRAM, WHICH CONTAINS THE INDEPENDENT
C*              VARIABLES. THESE VALUES MUST BE STRICTLY MONOTONIC.
C*
C*       NTAB   THE NUMBER OF DEPENDENT VARIABLE TABLES
C*
C*          Y   A TWO-DIMENSIONAL ARRAY DIMENSIONED (NMAX,NTAB) IN
C*              THE CALLING PROGRAM. EACH COLUMN OF THE ARRAY
C*              CONTAINS A DEPENDENT VARIABLE TABLE
C*
C*     IORDER   INTERPOLATION PARAMETER SUPPLIED BY THE USER.
C*
C*              =0  ZERO ORDER INTERPOLATION: THE FIRST FUNCTION
C*                  VALUE IN EACH DEPENDENT VARIABLE TABLE IS
C*                  ASSIGNED TO THE CORRESPONDING MEMBER OF THE Y0
C*                  ARRAY. THE FUNCTIONAL VALUE IS ESTIMATED TO
C*                  REMAIN CONSTANT AND EQUAL TO THE NEAREST KNOWN
C*                  FUNCTION VALUE.
C*
C*         X0   THE INPUT POINT AT WHICH INTERPOLATION WILL BE
C*              PERFORMED.
C*
C*         Y0   A ONE-DIMENSIONAL ARRAY DIMENSIONED (NTAB) IN THE
C*              CALLING PEOGRAM. UPON RETURN THE ARRAY CONTAINS THE
C*              ESTIMATED VALUE OF EACH FUNCTION AT X0.
C*
C*        IPT   ON THE FIRST CALL IPT MUST BE INITIALIZED TO -1 SO
C*              THAT MONOTONICITY WILL BE CHECKED. UPON LEAVING THE
C*              ROUTINE IPT EQUALS THE VALUE OF THE INDEX OF THE X
C*              VALUE PRECEDING X0 UNLESS EXTRAPOLATION WAS
C*              PERFORMED. IN THAT CASE THE VALUE OF IPT IS
C*              RETURNED AS:
C*              =0  DENOTES X0 .LT. X(1) IF THE X ARRAY IS IN
C*                  INCREASING ORDER AND X(1) .GT. X0 IF THE X ARRAY
C*                  IS IN DECREASING ORDER.
C*              =N  DENOTES X0 .GT. X(N) IF THE X ARRAY IS IN
C*                  INCREASING ORDER AND X0 .LT. X(N) IF THE X ARRAY
C*                  IS IN DECREASING ORDER.
C*
C*              ON SUBSEQUENT CALLS, IPT IS USED AS A POINTER TO
C*              BEGIN THE SEARCH FOR X0.
C*
C*       IERR   ERROR PARAMETER GENERATED BY THE ROUTINE
C*              =0  NORMAL RETURN
C*              =J  THE J-TH ELEMENT OF THE X ARRAY IS OUT OF ORDER
C*              =-1 ZERO ORDER INTERPOLATION PERFORMED BECAUSE
C*                  IORDER =0.
C*              =-2 ZERO ORDER INTERPOLATION PERFORMED BECAUSE ONLY
C*                  ONE POINT WAS IN X ARRAY.
C*              =-3 NO INTERPOLATION WAS PERFORMED BECAUSE
C*                  INSUFFICIENT POINTS WERE SUPPLIED FOR SECOND
C*                  ORDER INTERPOLATION.
C*              =-4 EXTRAPOLATION WAS PERFORMED
C*
C*              UPON RETURN THE PARAMETER IERR SHOULD BE TESTED IN
C*              THE CALLING PROGRAM.
```

```
C*                                                                    *IUNT0850
C*     REQUIRED ROUTINES                 NONE                         *IUNT0860
C*                                                                    *IUNT0870
C*     SOURCE                            CMPB ROUTINE MTLUP MODIFIED  *IUNT0880
C*                                       BY COMPUTER SCIENCES CORPORATION*IUNT0890
C*                                                                    *IUNT0900
C*     LANGUAGE                          FORTRAN                      *IUNT0910
C*                                                                    *IUNT0920
C*                                                                    *IUNT0930
C*     DATE RELEASED                     AUGUST 1,1973                *IUNT0940
C*                                                                    *IUNT0950
C*     LATEST REVISION                   JUNE 9, 1975                 *IUNT0960
C*                                                                    *IUNT0970
C**********************************************************************IUNT0980
      DIMENSION X(1),Y(NMAX,1),YO(1)                                   IUNT0990
      NM1=N-1                                                          IUNT1000
      IERR=0                                                           IUNT1010
      J=1                                                              IUNT1020
C                                                                      IUNT1030
C              TEST FOR ZERO ORDER INTERPOLATION                       IUNT1040
C                                                                      IUNT1050
      DELX=X(2)-X(1)                                                   IUNT1060
      IF (IORDER .EQ. 0) GO TO 10                                      IUNT1070
      IF (N.LT. 2) GO TO 20                                            IUNT1080
      GO TO 50                                                         IUNT1090
   10 IERR=-1                                                          IUNT1100
      GO TO 30                                                         IUNT1110
   20 IERR=-2                                                          IUNT1120
   30 DO 40 NT=1,NTAB                                                  IUNT1130
         YO(NT)=Y(1,NT)                                                IUNT1140
   40    CONTINUE                                                      IUNT1150
      RETURN                                                           IUNT1160
   50 IF (IPT .GT. -1) GO TO 65                                        IUNT1170
C                                                                      IUNT1180
C             CHECK FOR TABLE OF NODE POINTS BEING STRICTLY MONOTONIC  IUNT1190
C             THE SIGN OF DELX SIGNIFIES WHETHER TABLE IS IN           IUNT1200
C             INCREASING OR DECREASING ORDER.                          IUNT1210
C                                                                      IUNT1220
      IF (DELX .EQ. 0) GO TO 190                                       IUNT1230
      IF (N .EQ. 2) GO TO 65                                           IUNT1240
C                                                                      IUNT1250
C             CHECK FOR SIGN CONSISTENCY IN THE DIFFERENCES OF         IUNT1260
C             SUBSEQUENT PAIRS                                         IUNT1270
C                                                                      IUNT1280
      DO 60 J=2,NM1                                                    IUNT1290
         IF (DELX * (X(J+1)-X(J))) 190,190,60                          IUNT1300
   60    CONTINUE                                                      IUNT1310
C                                                                      IUNT1320
C             IPT IS INITIALIZED TO BE WITHIN THE INTERVAL             IUNT1330
C                                                                      IUNT1340
   65 IF (IPT .LT. 1) IPT=1                                            IUNT1350
      IF (IPT .GT. NM1) IPT=NM1                                        IUNT1360
      IN= SIGN (1.0,DELX *( X0-X(IPT)))                                IUNT1370
   70 P= X(IPT) - X0                                                   IUNT1380
      IF (P* (X(IPT +1)- X0)) 90,180,80                                IUNT1390
   80 IPT =IPT +IN                                                     IUNT1400
C                                                                      IUNT1410
C             TEST TO SEE IF IT IS NECCESARY TO EXTRAPOLATE            IUNT1420
C                                                                      IUNT1430
      IF (IPT.GT.0 .AND. IPT .LT. N) GO TO 70                          IUNT1440
      IERR=-4                                                          IUNT1450
      IPT=IPT- IN                                                      IUNT1460
C                                                                      IUNT1470
C             TEST FOR ORDER OF INTERPOLATION                          IUNT1480
C                                                                      IUNT1490
C                                                                      IUNT1500
```

```
 90   IF (IORDER .GT. 1) GO TO 120
C
C             FIRST ORDER INTERPOLATION
C
      IPT1=IPT+1
      XTMP1=X0-X(IPT)
      XTMP2=X(IPT1)-X(IPT)
      XTMP1=XTMP1/XTMP2
      DO 100 NT=1,NTAB
         YTMP=Y(IPT1,NT)-Y(IPT,NT)
         Y0(NT)=Y(IPT,NT)+YTMP*XTMP1
100      CONTINUE
      IF (IERR .EQ. -4) IPT=IPT+IN
      RETURN
C
C             SECOND ORDER INTERPOLATION
C
120   IF (N .EQ. 2) GO TO 200
C
C             CHOOSING A THIRD POINT SO AS TO MINIMIZE THE DISTANCE
C             BETWEEN THE THREE POINTS USED TO INTERPOLATE
C
      IF (IPT .EQ. NM1) GO TO  140
      IF (IPT .EQ. 1) GO TO 130
      A1=ABS(X0-X(IPT-1))
      A2=ABS(X(IPT+2)-X0)
      IF(A1-A2)140,130,130
130   L=IPT
      GO TO 150
140   L=IPT -1
150   V1=X(L)-X0
      V2=X(L+1)-X0
      V3=X(L+2)-X0
      DO 160 NT=1,NTAB
      YY1=(Y(L,NT) * V2 - Y(L+1,NT) * V1)/(X(L+1) - X(L))
      YY2=(Y(L+1,NT)*V3-Y(L+2,NT) *V2)/(X(L+2)-X(L+1))
      Y0(NT)=(YY1*V3-YY2*V1)/(X(L+2)-X(L))
 160  CONTINUE
      IF (IERR .EQ. -4) IPT=IPT + IN
      RETURN
180   IF(P .NE. 0) IPT=IPT +1
      DO 185 NT=1,NTAB
         Y0(NT)=Y(IPT,NT)
185      CONTINUE
      RETURN
C
C             IERR IS SET TO THE SUBSCRIPT OF THE MEMBER OF THE TABLE
C             WHICH IS OUT OF ORDER
C
190   IERR=J +1
      RETURN
200   IERR=-3
      RETURN
      END
```

REFERENCES

1. Reubush, David E.; and Putnam, Lawrence E.: An Experimental and Analytical Investigation of the Effect on Isolated Boattail Drag of Varying Reynolds Number up to 130×10^6. NASA TN D-8210, 1976.

2. Wilmoth, Richard G.: Computation of Transonic Boattail Flow With Separation. NASA TP-1070, 1977.

3. Chow, Wen L.; Bober, Lawrence J.; and Anderson, Bernhard H.: Numerical Calculation of Transonic Boattail Flow. NASA TN D-7984, 1975.

4. Kuhn, Gary D.: Computer Program for Calculation of Separated Turbulent Flows on Axisymmetric Afterbodies. AEDC-TR-77-72, U.S. Air Force, June 1977. (Available from DDC as AD A041 219.)

5. Yaros, S. F.: Prediction of Pressure Distributions on Axisymmetric Bodies in Transonic Flow. AIAA Paper No. 77-226, Jan. 1977.

6. Cosner, Raymond R.; and Bower, William W.: A Patched Solution of the Transonic Flowfield About an Axisymmetric Boattail. AIAA Paper 77-227, Jan. 1977.

7. Yaeger, L. S.: Transonic Flow Over Afterbodies Including the Effects of Jet-Plume and Viscous Interactions With Separation. AIAA Paper No. 77-228, Jan. 1977.

8. Presz, Walter Michael, Jr.: Turbulent Boundary Layer Separation of Axisymmetric Afterbodies. Ph. D. Thesis, Univ. of Connecticut, 1974.

9. Presz, Walter M., Jr.; and Pitkin, Edward T.: Flow Separation Over Axisymmetric Afterbody Models. AIAA Paper No. 74-17, Jan.-Feb. 1974.

10. Presz, Walter M., Jr.; King, Ronald W.; and Buteau, John D.: An Improved Analytical Model of the Separated Region on Boattail Nozzles at Subsonic Speeds. NASA CR-3028, 1978.

11. Presz, Walter M., Jr.; King, Ronald W.; Buteau, John D.; and Putnam, Lawrence E.: An Improved Analytical Model of the Separated Region on Nozzle Boattails. AIAA Paper 78-995, July 1978.

12. Hess, J. L.; and Smith, A. M. O.: Calculation of Potential Flow About Arbitrary Bodies. Progress in Aeronautical Sciences, Volume 8, D. Küchemann, P. Carrière, B. Etkin, W. Fiszdon, N. Rott, J. Smolderen, I. Tani, and W. Wuest, eds., Pergamon Press, Inc., c.1967, pp. 1-138.

13. Labrujere, Th. E.; Loeve, W.; and Slooff, J. W.: An Approximate Method for the Calculation of the Pressure Distribution on Wing-Body Combinations at Subcritical Speeds. Aerodynamic Interference, AGARD-CP-71-71, Jan. 1971, pp. 11-1 - 11-15.

14. Salas, Manuel D.: The Numerical Calculation of Inviscid Plume Flow Fields. AIAA Paper No. 74-523, June 1974.

15. Users Manual for the External Drag and Internal Nozzle Performance Deck (Deck XI) - Supersonic Flow Analysis (Applicable to Deck VI). PWA-3465, Suppl. F, Pt. I (Contract No. AF33(615)-3128), Pratt & Whitney Aircraft, Sept. 1, 1968.

16. Reshotko, Eli; and Tucker, Maurice: Approximate Calculation of the Compressible Turbulent Boundary Layer With Heat Transfer and Arbitrary Pressure Gradient. NACA TN 4154, 1957.

17. Winter, K. G.; Rotta, J. C.; and Smith, K. G.: Studies of the Turbulent Boundary Layer on a Waisted Body of Revolution in Subsonic and Supersonic Flow. R. and M. No. 3633, British A.R.C., 1970.

18. Goldschmied, Fabio R.: An Approach to Turbulent Incompressible Separation and the Determination of Skin-Friction Under Adverse Pressure Gradients. AIAA Paper No. 64-465, June-July 1964.

19. Page, R. H.: A Theory for Incipient Separation. Developments in Mechanics, Volume 1, J. E. Lay and L. E. Malvern, eds., Plenum Press, 1961, pp. 563-577.

20. Stratford, B. S.: The Prediction of Separation of the Turbulent Boundary Layer. J. Fluid Mech., vol. 5, pt. 1, Jan. 1959, pp. 1-16.

21. Abeyounis, William Kelly: Boundary-Layer Separation on Isolated Boattail Nozzles. NASA TP-1226, 1978.

22. Putnam, Lawrence E.; and Abeyounis, William K.: Experimental and Theoretical Study of Flow Fields Surrounding Boattail Nozzles at Subsonic Speeds. AIAA Paper No. 76-675, July 1976.

23. Reubush, David E.: Experimental Study of the Effectiveness of Cylindrical Plume Simulators for Predicting Jet-On Boattail Drag at Mach Numbers up to 1.30. NASA TN D-7795, 1974.

24. Reubush, David E.; and Runckel, Jack F.: Effect of Fineness Ratio on the Boattail Drag of Circular-Arc Afterbodies Having Closure Ratios of 0.50 With Jet Exhaust at Mach Numbers Up to 1.30. NASA TN D-7192, 1973.

25. Berrier, Bobby L.: Effect of Nonlifting Empennage Surfaces on Single-Engine Afterbody/Nozzle Drag at Mach Numbers From 0.5 to 2.2. NASA TN D-8326, 1977.

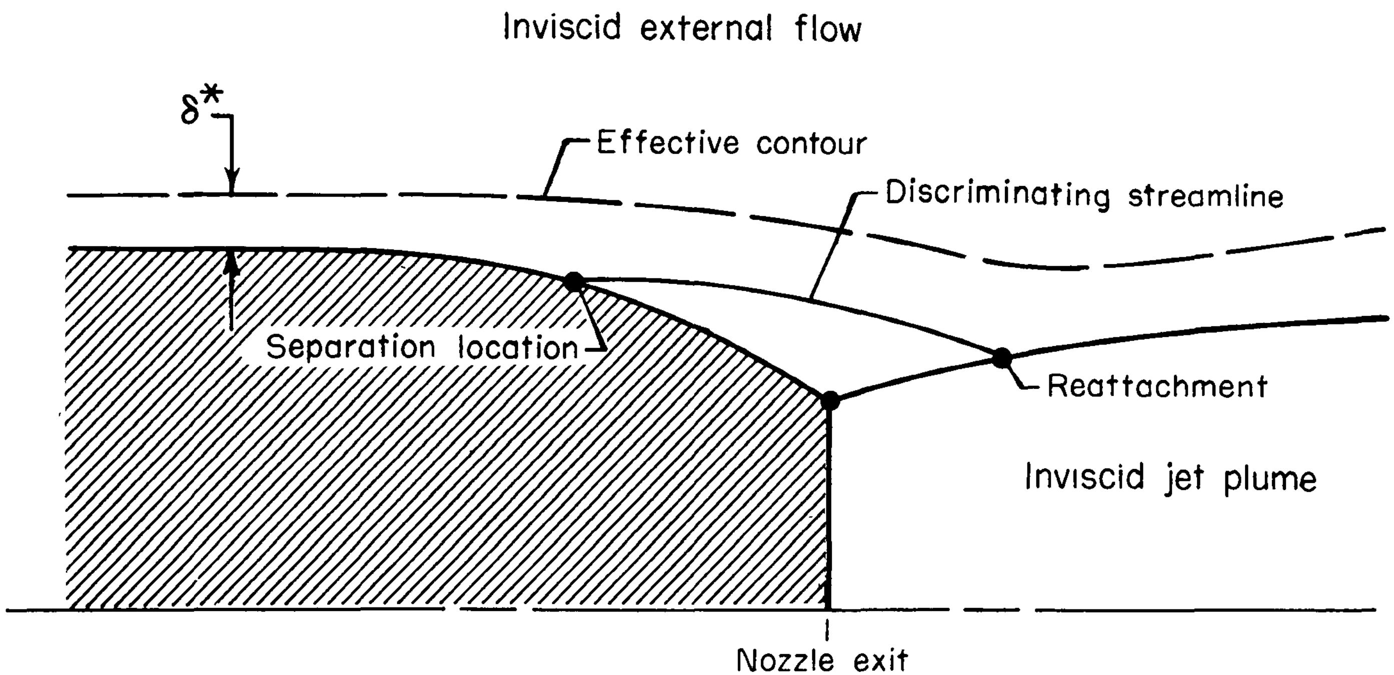


Figure 1.- Analytical model of flow over nozzle boattail.

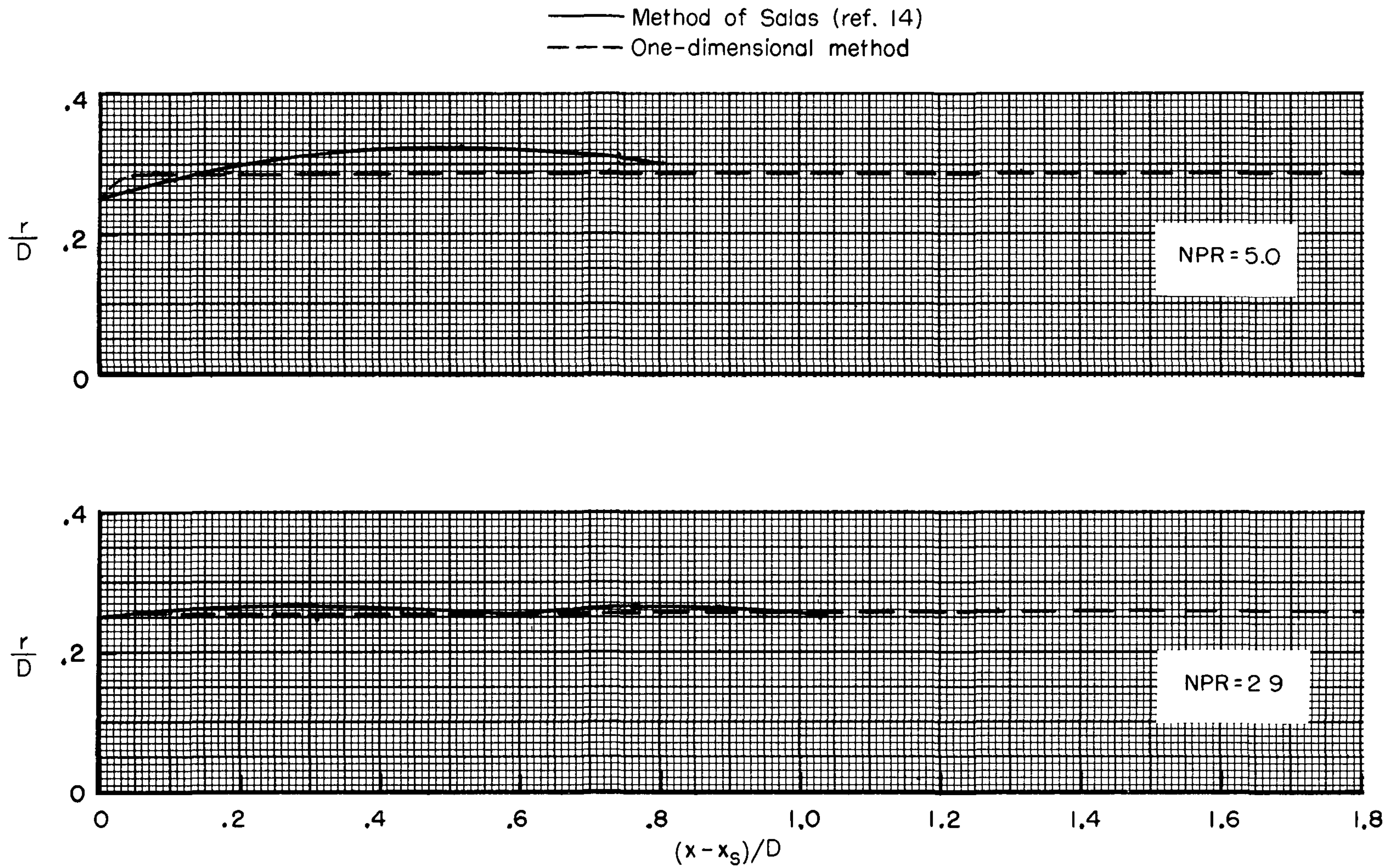
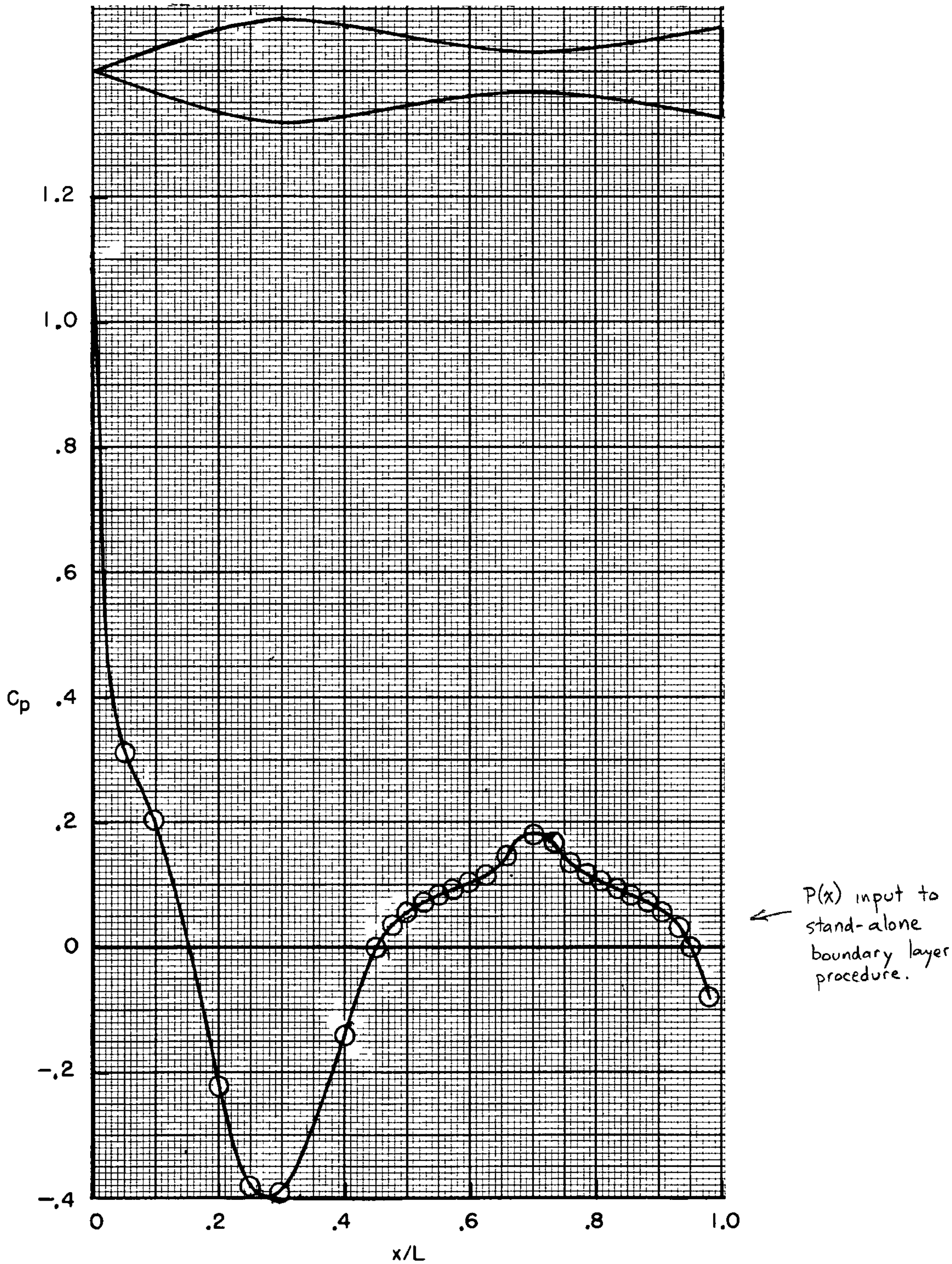
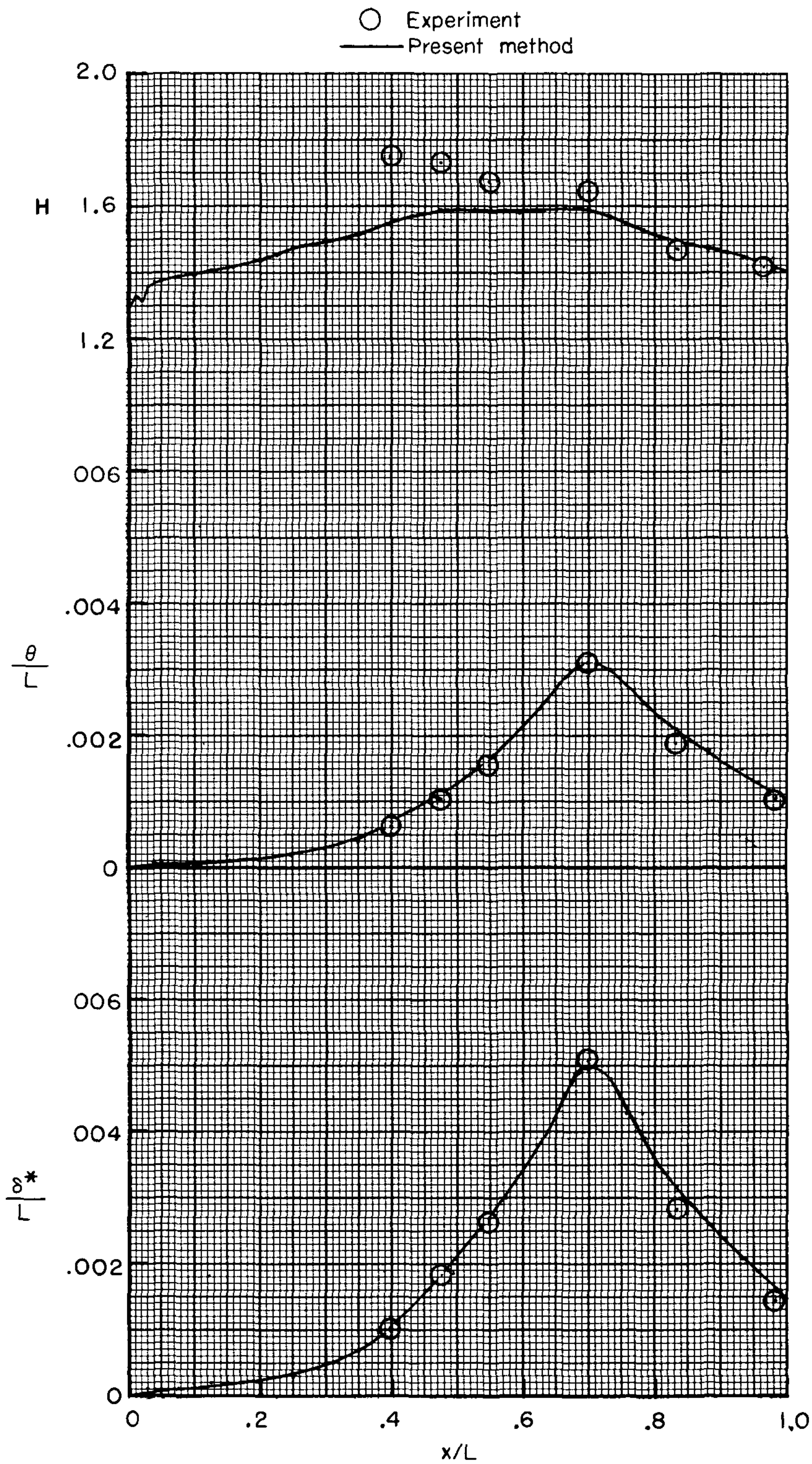


Figure 2.- Comparison of one-dimensional jet exhaust flow calculation with method of Salas (ref. 14).



(a) Pressure distribution and body geometry.

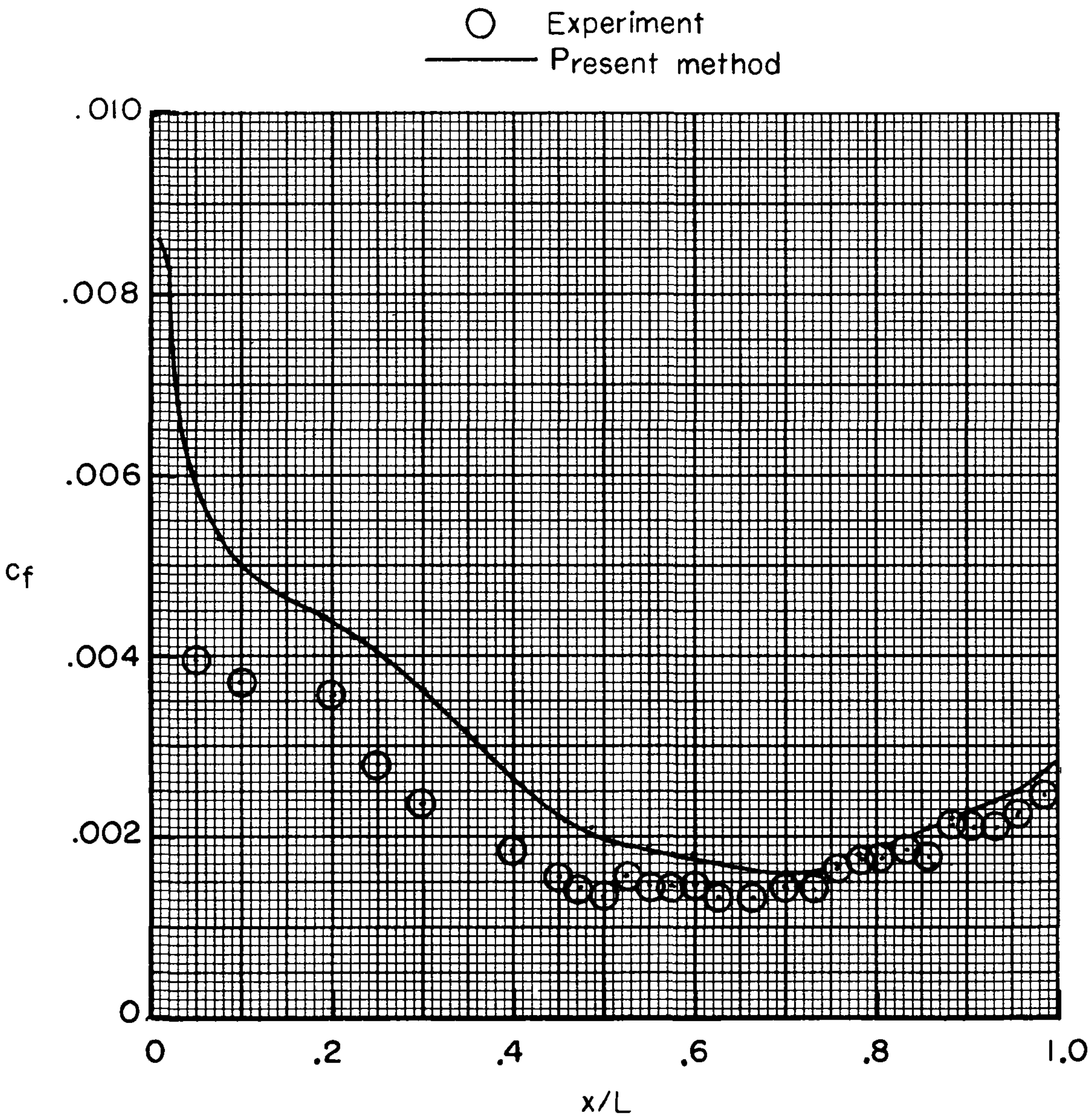
Figure 3.- Comparison of predicted boundary-layer characteristics with experiment of Winter, Rotta, and Smith (ref. 17). $M_\infty = 0.6$ and Reynolds number based on body length of 9.85×10^6.



(b) Displacement thickness, momentum thickness, and shape factor.

Figure 3.- Continued.

These are results from a stand-alone boundary-layer procedure (also next page).



(c) Skin friction.

Figure 3.- Concluded.

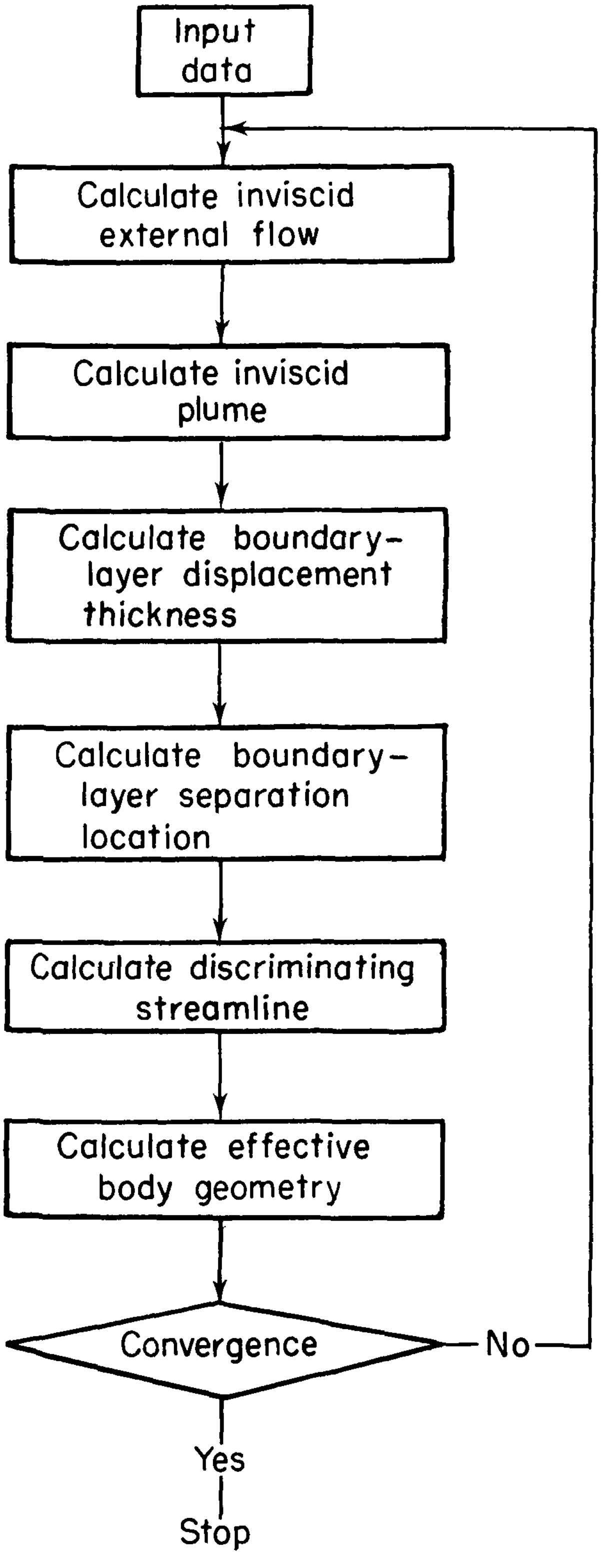


Figure 4.- Flow diagram of interaction procedure.

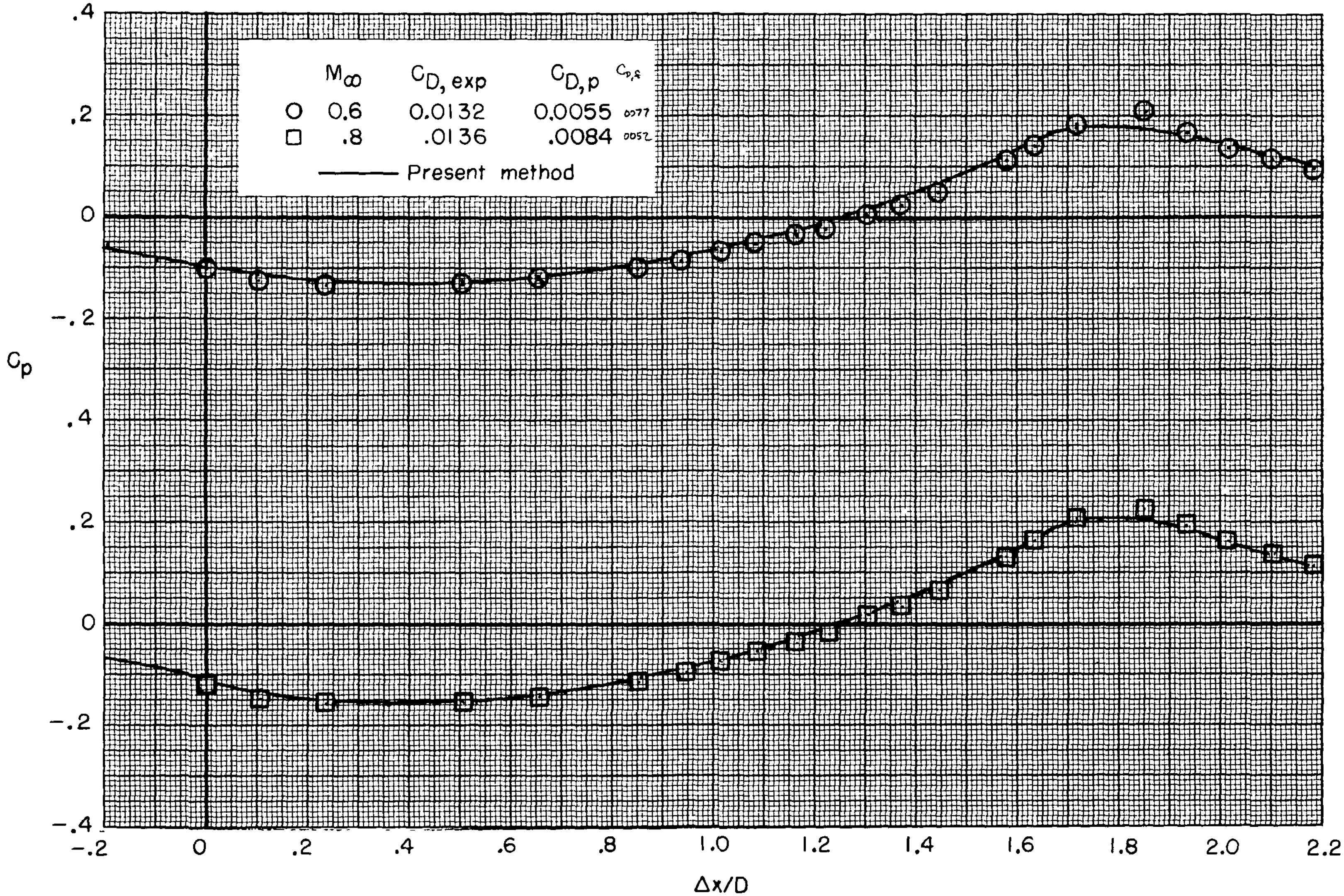
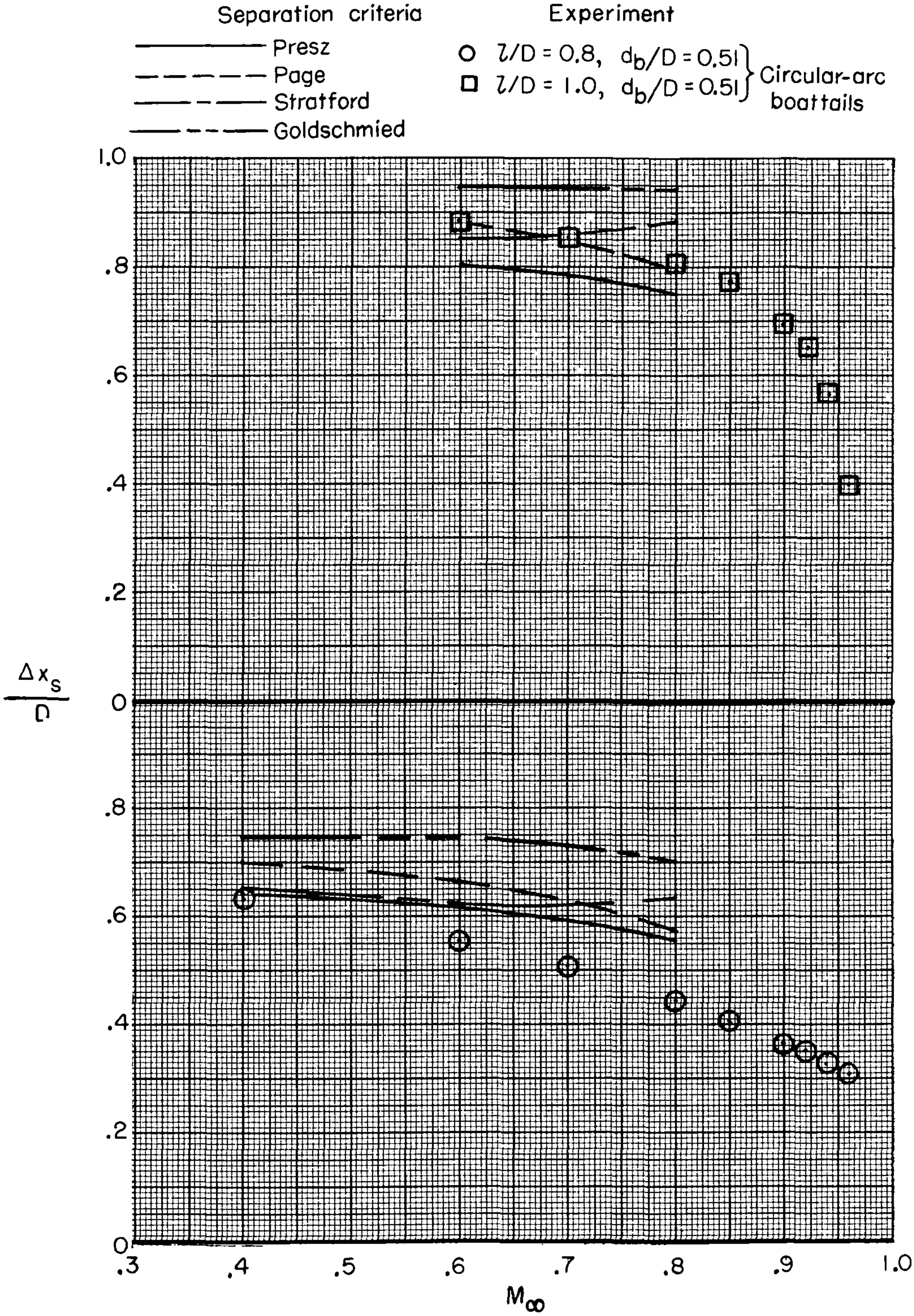


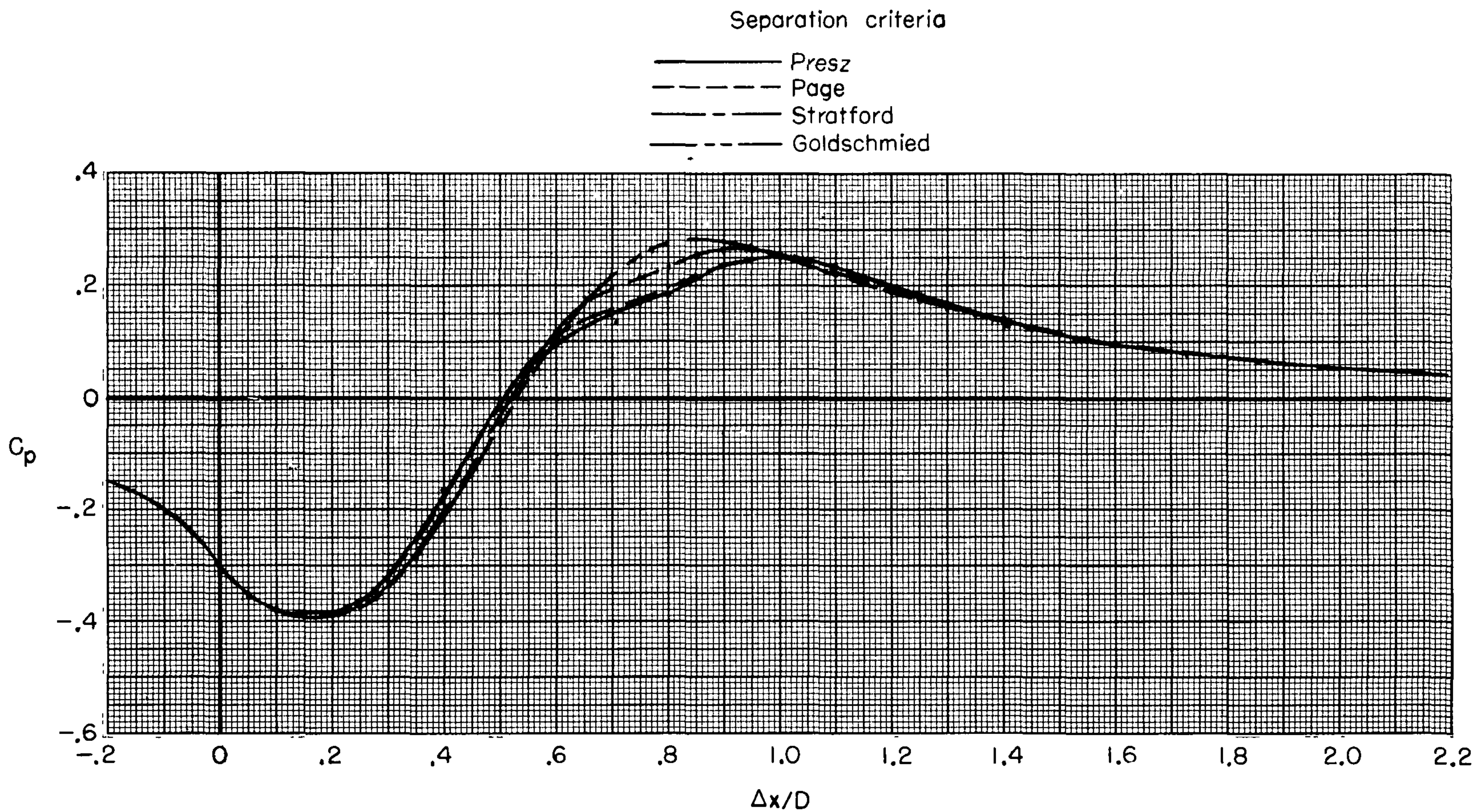
Figure 5.- Comparison of theory and experiment for flow over unseparated $l/D = 1.768$, $d_b/D = 0.51$ circular-arc nozzle with solid jet plume simulator. (Experimental data from ref. 22.)

TN-D-7192
Config #3



(a) Separation location.

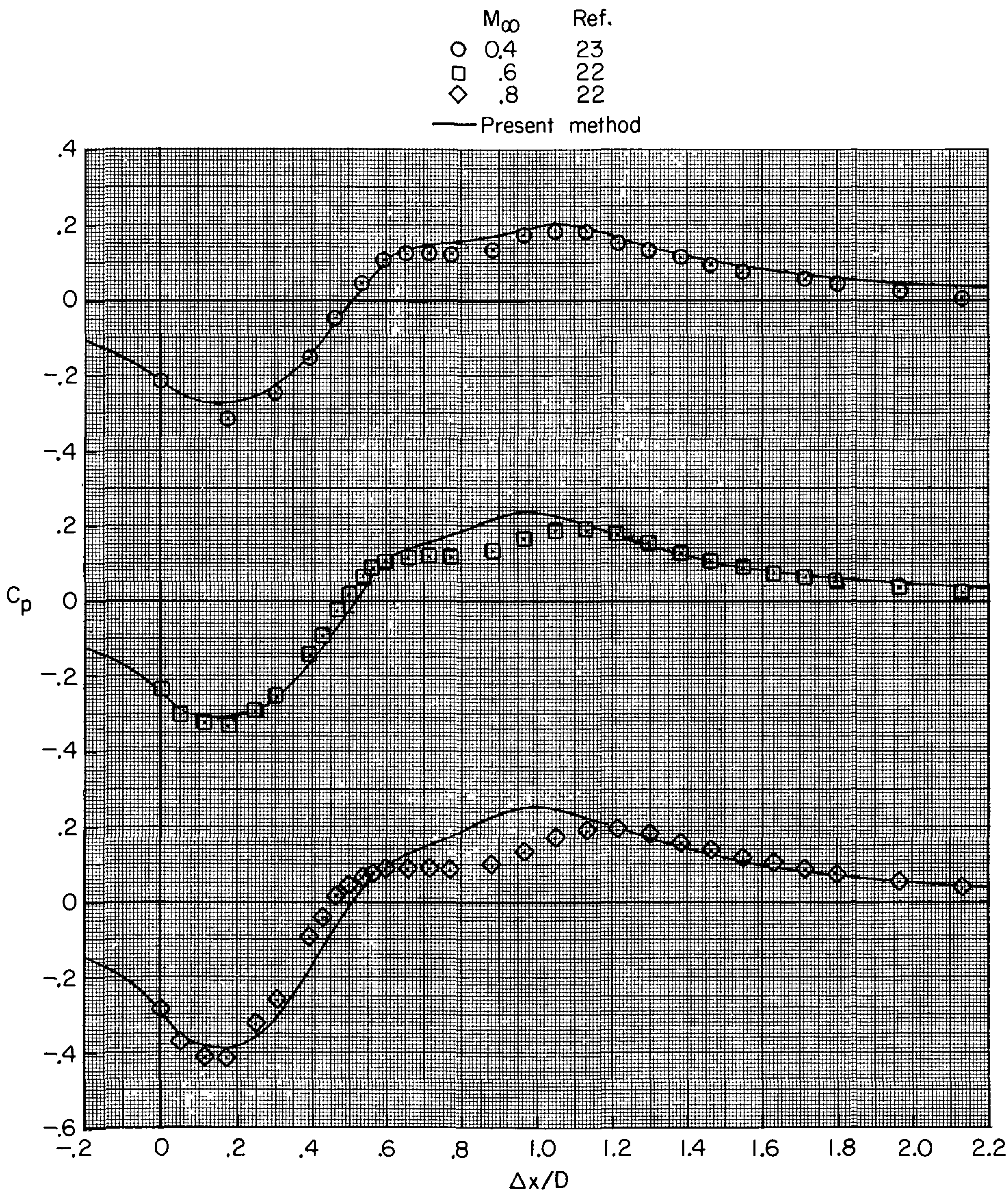
Figure 6.- Effect of separation location criteria. (Experimental data from ref. 21.)



(b) Predicted pressure distributions on $l/D = 0.8$, $d_b/D = 0.51$ circular-arc boattail at $M_\infty = 0.8$.

Figure 6.- Concluded.

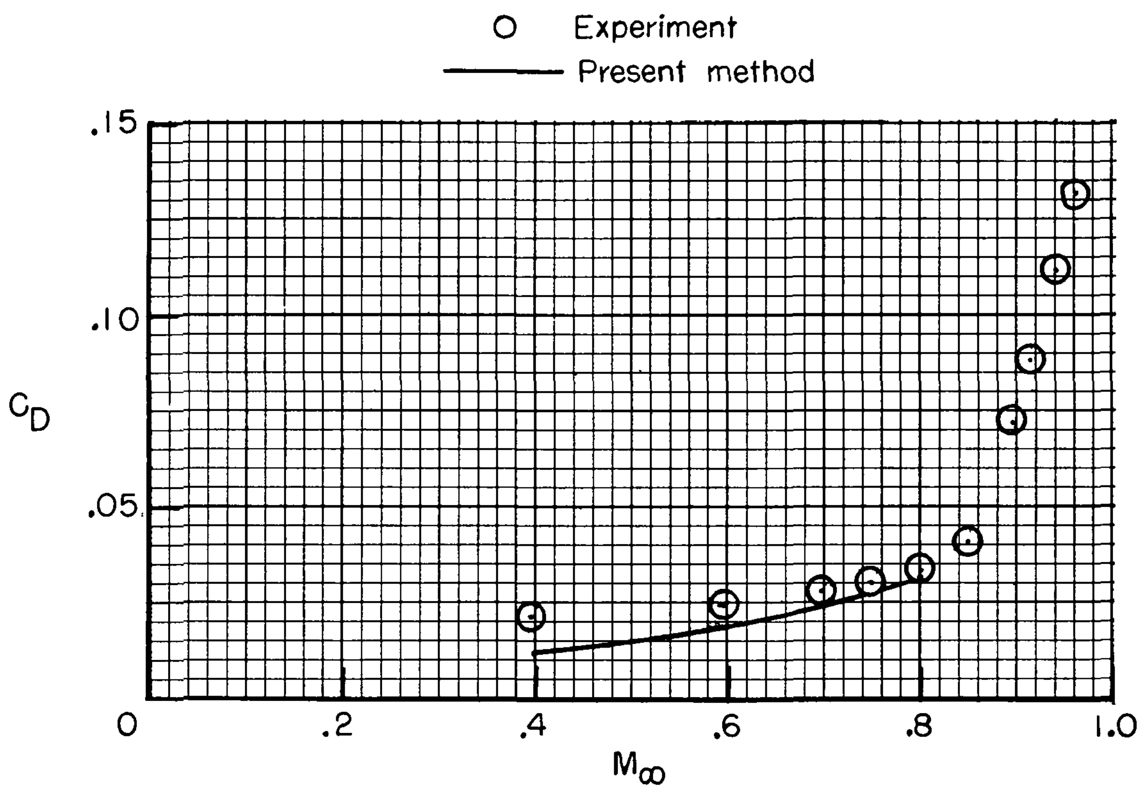
- you can get any pressure drag you want by altering this assumption!!



(a) Pressure distribution.

Figure 7.- Effect of Mach number on flow over $l/D = 0.8$, $d_b/D = 0.51$ circular-arc boattail with solid jet plume simulator.

Langley #1



(b) Pressure drag coefficient.

Figure 7.- Concluded.

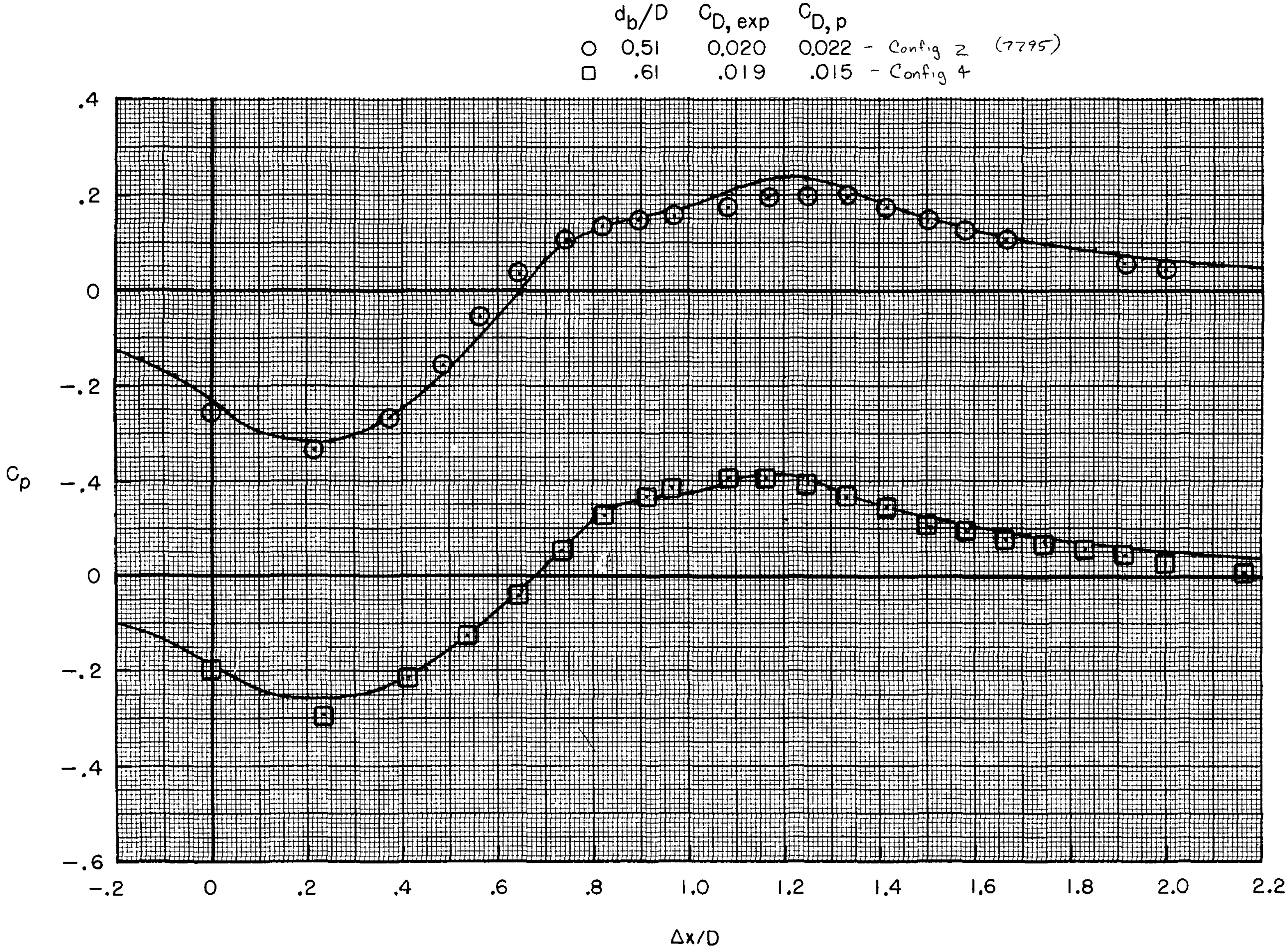
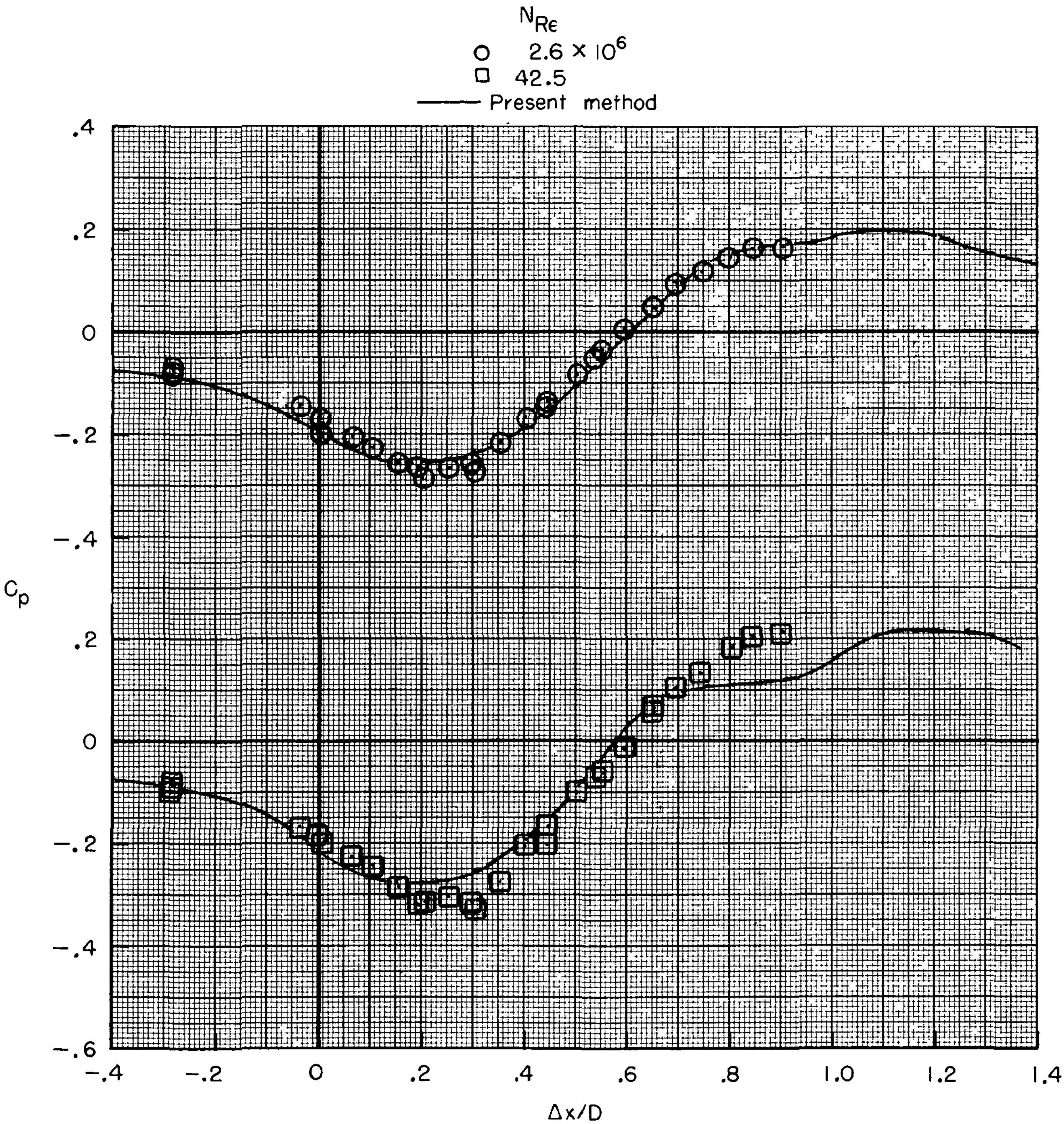


Figure 8.- Effect of afterbody closure on flow over $l/D = 1.0$ circular-arc boattails with solid jet plume simulators. $M_\infty = 0.8$. (Experimental data from ref. 23.)

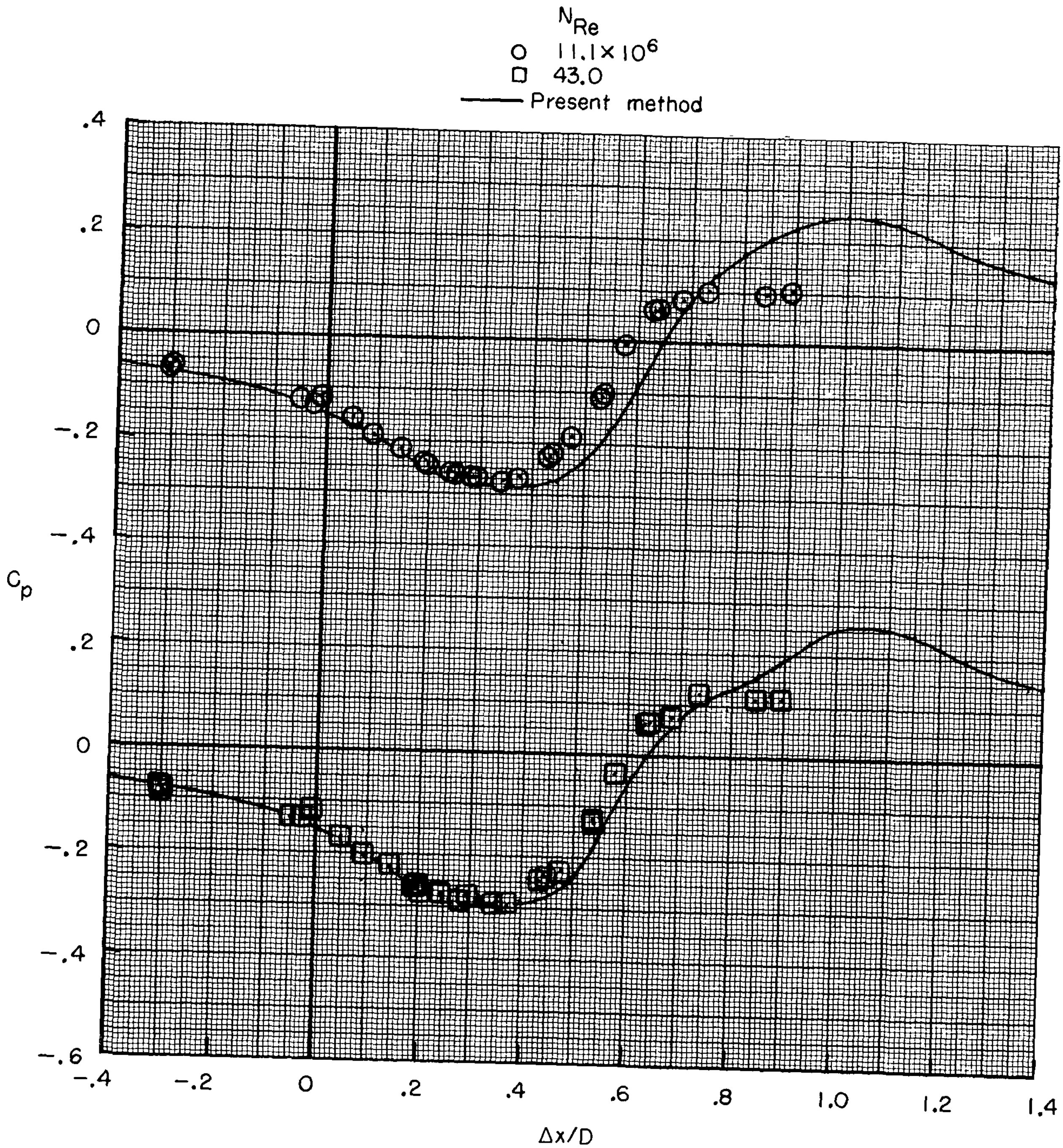
TN-D-7795



(a) Pressure distributions for $l/D = 0.961$, $d_b/D = 0.51$ circular-arc conic boattail.

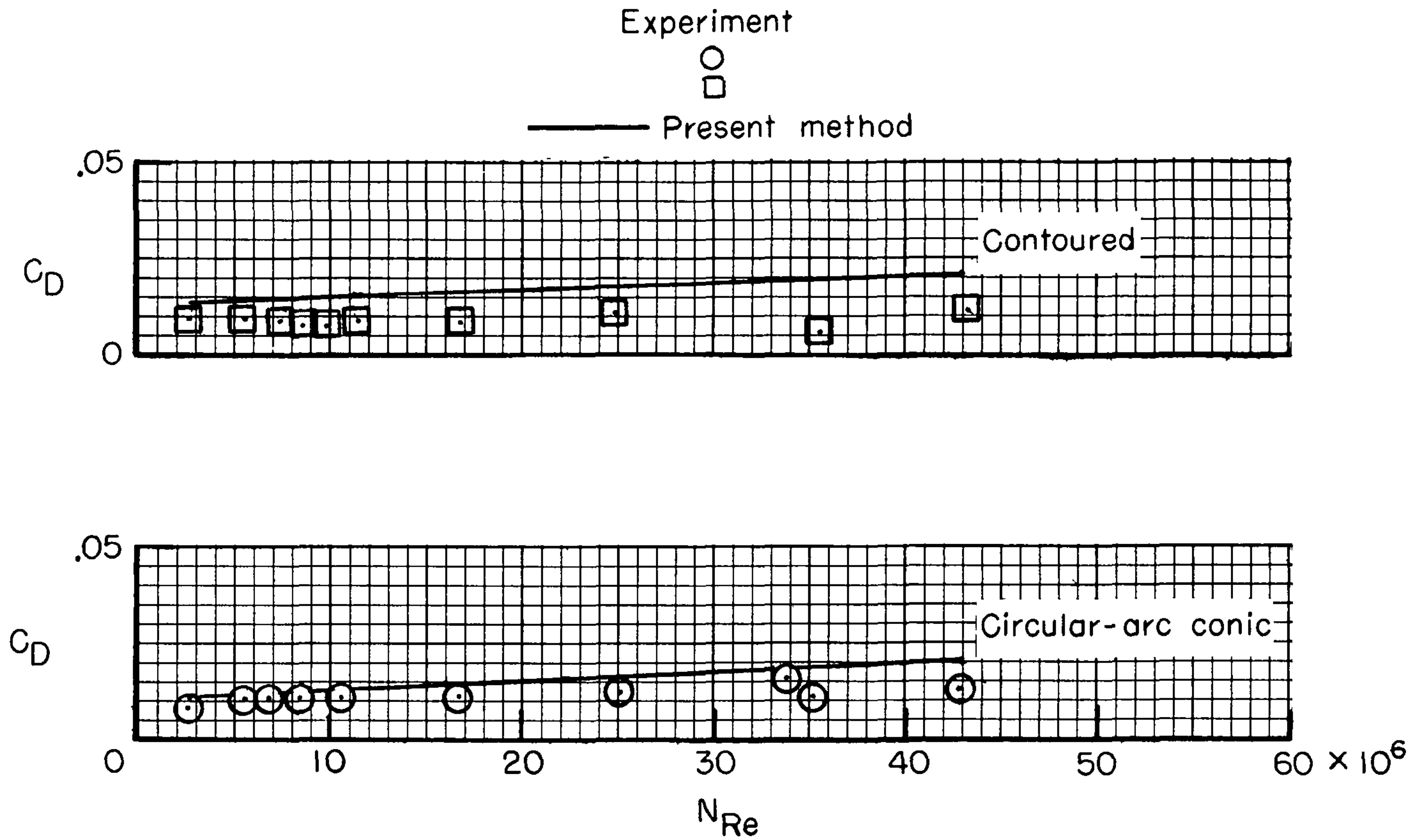
Figure 9.- Effect of Reynolds number on pressures and drag of afterbodies with solid jet plume simulators. $M_\infty = 0.6$. (Experimental data from ref. 1.)

TN-D-8210
"Circular-arc conic"



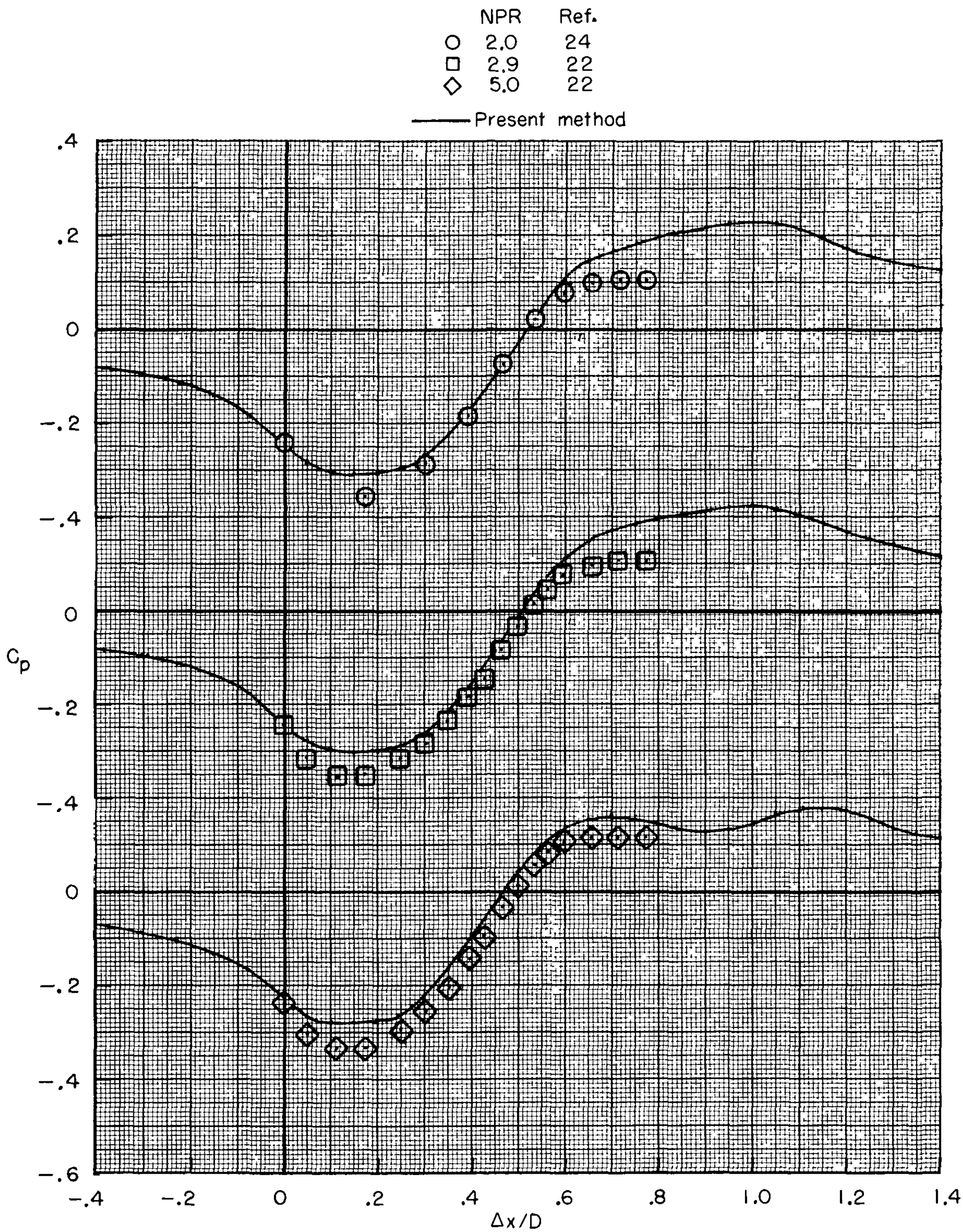
(b) Pressure distributions for $l/D = 0.95$, $d_b/D = 0.544$ contoured boattail.

Figure 9.- Continued.



(c) Drag coefficient.

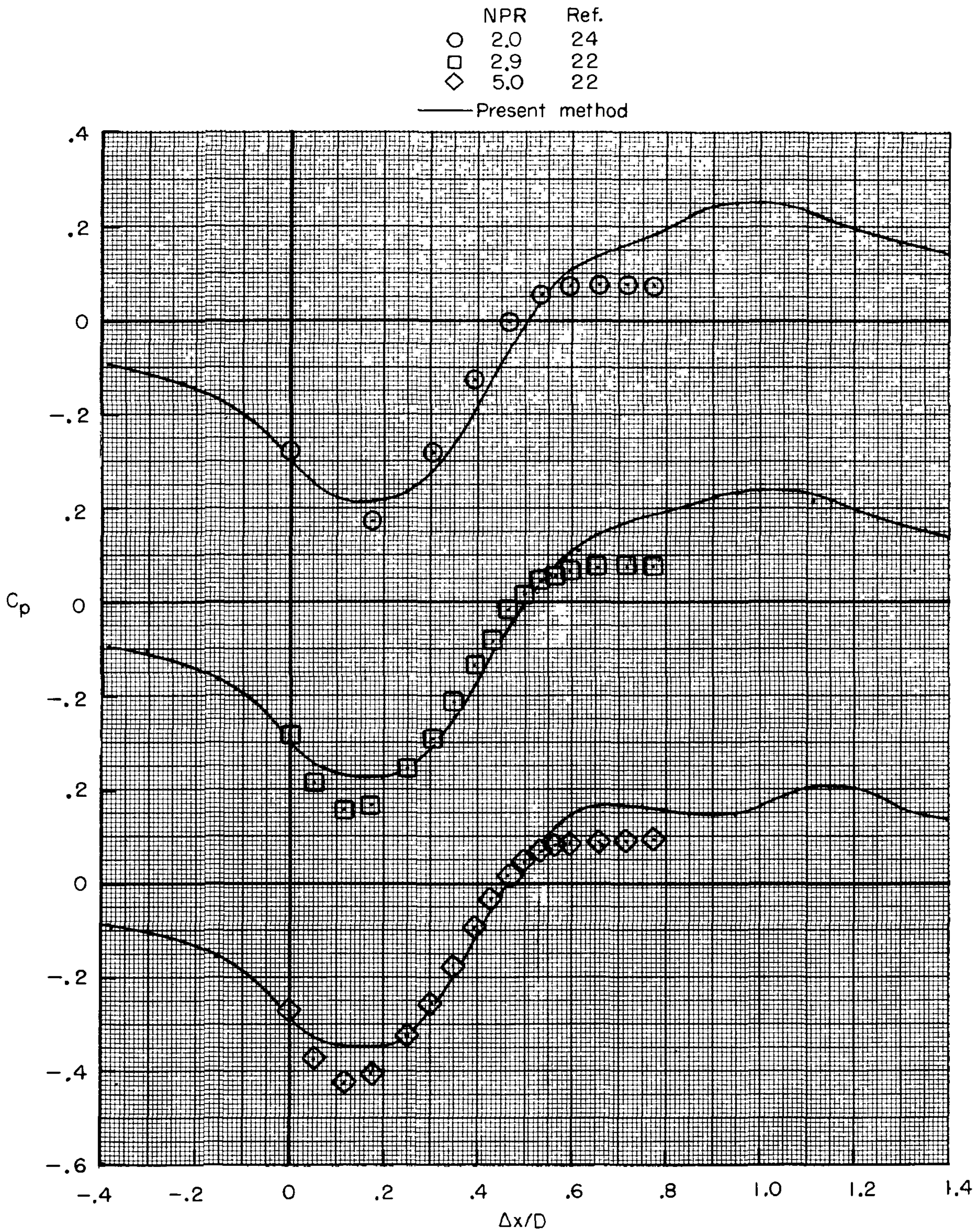
Figure 9.- Concluded.



(a) Pressure distribution at $M_\infty = 0.6$.

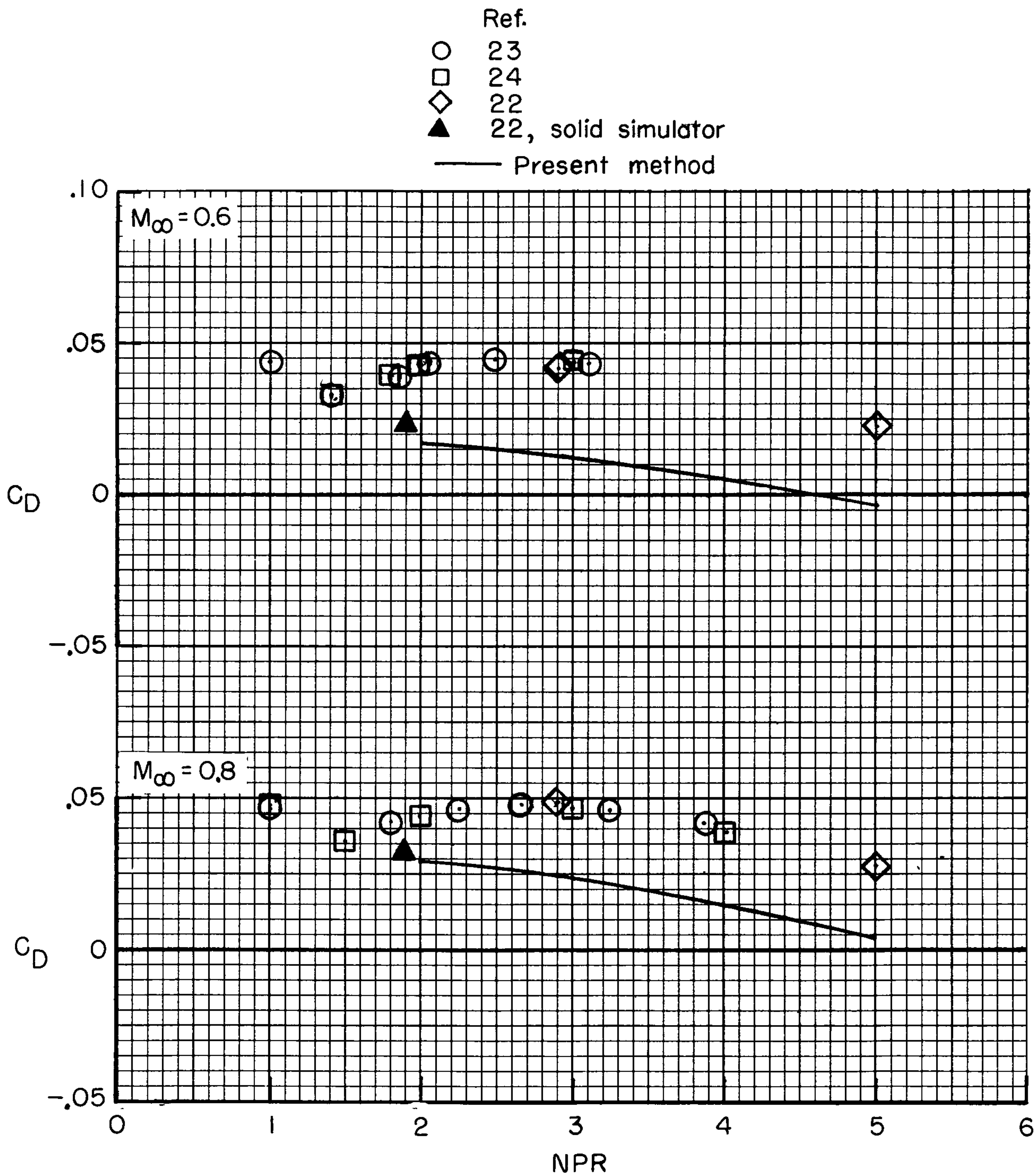
Figure 10.- Effect of NPR on pressures and drag for $l/D = 0.8$, $d_b/D = 0.51$ circular-arc nozzle.

prediction is far more sensitive to NPR than the data



(b) Pressure distribution at $M_\infty = 0.8$.

Figure 10.- Continued.



(c) Drag coefficient.

Figure 10.- Concluded.

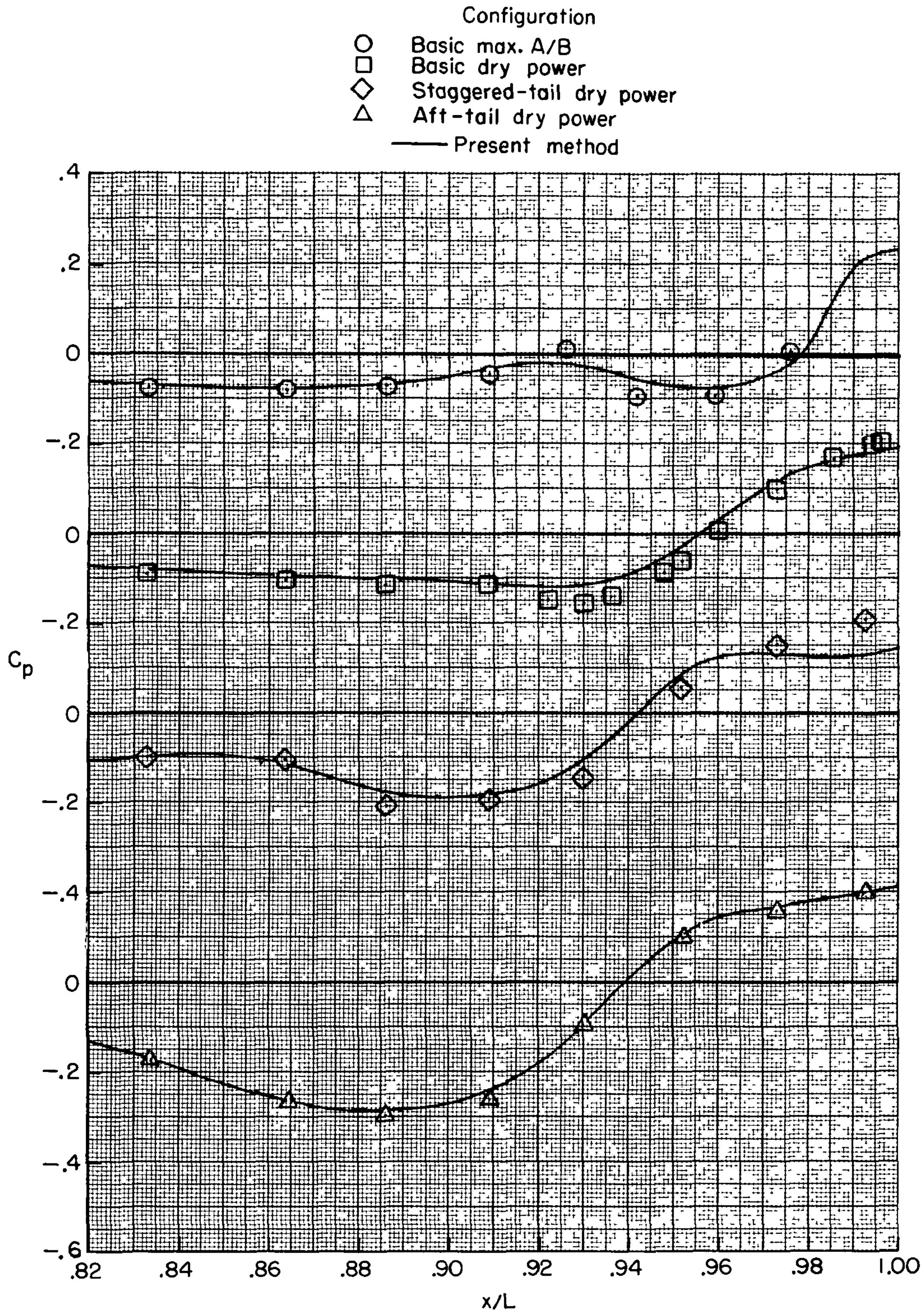


Figure 11.- Comparison of experimental and predicted pressures for equivalent bodies of Berrier (ref. 25). $M_\infty = 0.8$ and NPR = 2.5.

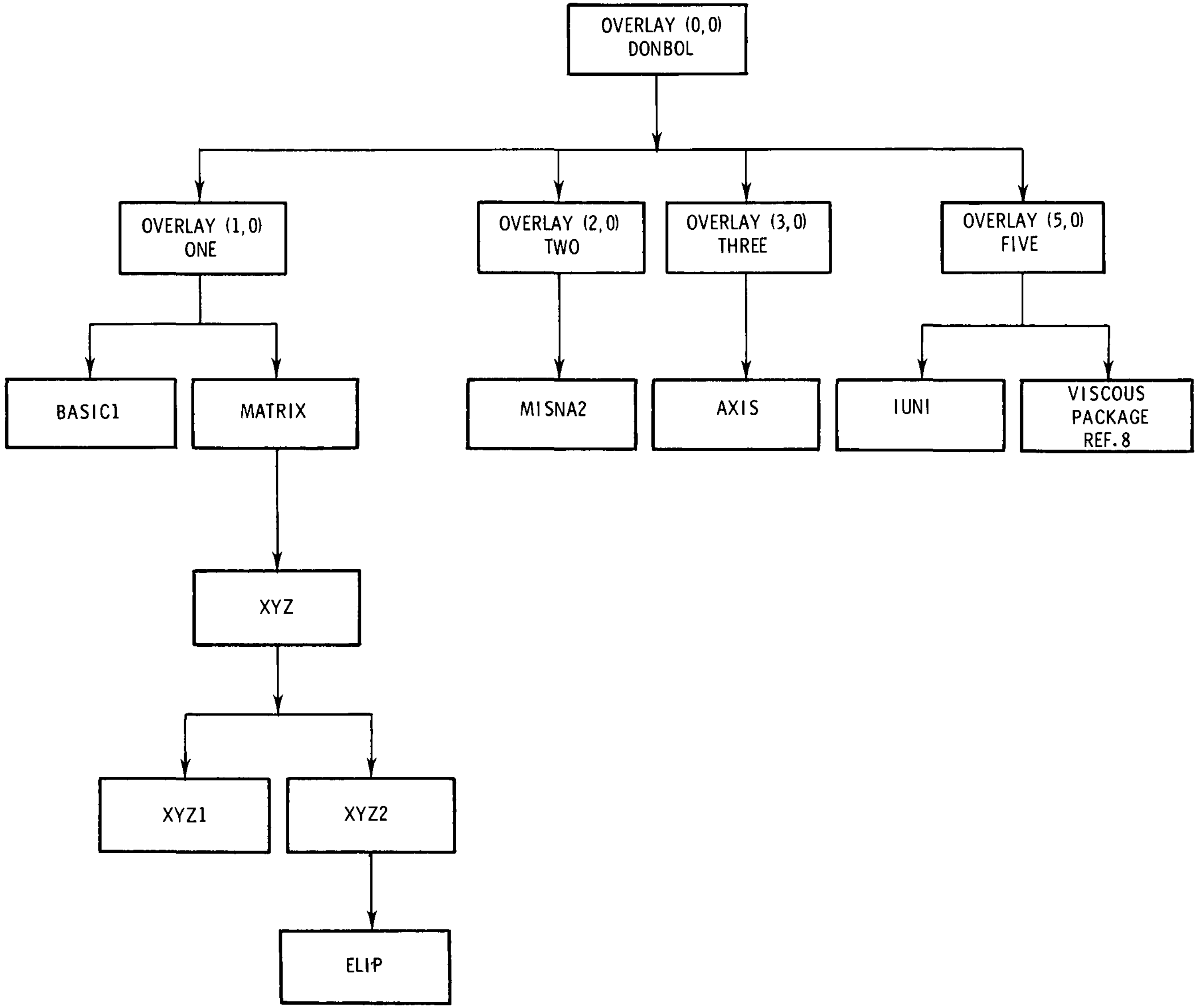


Figure 12.- Flow chart for program DONBOL.

```
**TEST CASE**  L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE  NPR=5.03
   2   15    1    0    15    2    2    1  113  135    1  135    1    1
0.80        100720.0  324.44     0.1524     0.0182415
1.000       332342.7  295.56     0.250
  167
   0.000000   .076271    .152542    .228814    .305085    .381356
    .457627   .533898    .610169    .686441    .762712    .838983
    .915254   .991525   1.067797   1.144068   1.220339   1.296610
   1.372881  1.449153   1.525424   1.601695   1.677966   1.754237
   1.830508  1.906780   1.983051   2.059322   2.135593   2.211864
   2.288136  2.364407   2.440678   2.516949   2.593220   2.669492
   2.745763  2.822034   2.898305   2.974576   3.050847   3.127119
   3.203390  3.279661   3.355932   3.432203   3.508475   3.584746
   3.661017  3.737288   3.813559   3.889831   3.966102   4.042373
   4.118644  4.194915   4.271186   4.347458   4.423729   4.500000
   4.576271  4.652542   4.728814   4.805085   4.881356   4.957627
   5.033898  5.110169   5.186441   5.262712   5.338983   5.415254
   5.491525  5.567797   5.644068   5.720339   5.796610   5.872881
   5.949153  6.025424   6.101695   6.177966   6.254237   6.330508
   6.406780  6.483051   6.559322   6.635593   6.711864   6.788136
   6.864407  6.940678   7.016949   7.093220   7.169492   7.245763
   7.322034  7.398305   7.474576   7.550847   7.627119   7.703390
   7.779661  7.855932   7.932203   8.008475   8.084746   8.161017
   8.237288  8.313559   8.389831   8.466102   8.542373   8.618644
   8.694915  8.771186   8.847458   8.923729   9.000000   9.050000
   9.100000  9.150000   9.200000   9.250000   9.300000   9.350000
   9.400000  9.450000   9.500000   9.550000   9.600000   9.650000
   9.700000  9.750000   9.800000   9.850000   9.900000   9.950000
  10.000000 10.050000  10.100000  10.150000  10.200000  10.250000
  10.300000 10.350000  10.400000  10.450000  10.500000  10.550000
  10.600000 10.664562  10.729125  10.793687  10.858250  10.922812
  10.987375 11.051937  11.116500  11.181062  11.245625  11.310187
  11.374750 11.439312  11.503875  11.568437  11.633000
   0.000000   .019011    .038021    .057032    .076043    .095053
    .114064   .133075    .152085    .171096    .190107    .209117
    .228128   .247139    .265867    .283850    .301088    .317584
    .333344   .348371    .362670    .376244    .389097    .401232
    .412653   .423362    .433361    .442654    .451243    .459130
    .466317   .472806    .478598    .483695    .488099    .491809
    .494827   .497155    .498791    .499738    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .500000
    .500000   .500000    .500000    .500000    .500000    .499125
    .496496   .492104    .485931    .477956    .468146    .456463
    .442859   .427277    .409646    .389885    .367897    .343565
    .316754   .287298    .255000    .250000    .250000    .250000
    .250000   .250000    .250000    .250000    .250000    .250000
    .250000   .250000    .250000    .250000    .250000    .250000
    .250000   .250000    .250000    .250000    .250000    .250000
    .250000   .250000    .250000    .250000    .250000    .250000
    .250000   .250000    .250000    .250000    .250000
   10
   8.800000  9.000000   9.200000   9.400000   9.600000   9.800000
   9.800000  9.800000   9.800000   9.800000
   0.60000    .600000    .600000    .600000    .600000   .600000
    .550000   .500000    .450000    .400000
```

Figure 13.- Sample input data

```
DONBOL  ***  AN AXISYMMETRIC INVISCID/VISCID INTERACTION PROGRAM

      BY LAWRENCE E. PUTNAM, NASA, LANGLEY RESEARCH CENTER

CASE TITLE = **TEST CASE**  L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE  NPR=5.03

     ***** CASE CONTROL DATA *****


     OFF-BODY POINTS
     LABRUJERE COMPRESSIBILITY CORRECTION
     MODIFIED RESHOTKO TUCKER BOUNDARY LAYER SOLUTION
     PRESZ MODIFIED CONTROL VOLUME DISCRIMINATING STREAMLINE SOLUTION
     PRESZ CONTROL VOLUME SEPARATION LOCATION CRITERIA
     START SEARCH FOR SEPARATION AT I = 113
     END SEARCH FOR SEPARATION AT I = 135
     JET EXHAUST PLUME CALCULATION
     NOZZLE EXIT AT I = 135
     SMOOTH AERODYNAMIC CONTOUR
     SMOOTH PRESSURE DISTRIBUTION

     FREE STREAM CONDITIONS
            MACH NUMBER       =          .800
            TOTAL PRESSURE    =    100720.000 PASCALS
            TOTAL TEMPERATURE =       324.440 KELVIN
            REYNOLDS NUMBER   =        12.182 MILLION PER METER

     JET EXHAUST CONDITIONS AT NOZZLE EXIT
            MACH NUMBER       =         1.000
            TOTAL PRESSURE    =    332342.700 PASCALS
            TOTAL TEMPERATURE =       295.560 KELVIN
            NPR               =         5.030
```

(a) Page 1.

Figure 14.- Sample output data.

```
FOR ITERA=  0    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  1    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  2    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  3    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  4    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  5    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  6    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  7    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  8    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA=  9    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 10    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 11    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 12    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 13    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 14    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
FOR ITERA= 15    8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2
```

(b) Page 2.

Figure 14.- Continued.

TEST CASE L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE NPR=5.03 ITERATION NO 15

MO = .8000 TT = 324.44 KELVIN PT = 100720.0 PASCALS L = .152400 METERS SREF = .018242 SQ METERS

BOUNDARY LAYER SEPARATION AT X/L = 9.529286 BOUNDARY LAYER REATTACHMENT AT X/L = 11.633000

X/L	R/L	CP	CF	CDP	CDF	CDT	RDS/L	RC/L	DEL*/L	DEL/L	THETA/L	H
0.0000	0.0000	1.1704	.0095	0.0000	0.0000	0.0000	0.0000	.0001	0.0000	0.0000	0.0000	0.0000
.0763	.0190	.4430	.0095	.0006	.0000	.0007	.0190	.0191	.0001	.0010	.0001	1.3777
.1525	.0380	.2840	.0067	.0021	.0001	.0022	.0380	.0383	.0003	.0023	.0002	1.4627
.2288	.0570	.2635	.0058	.0041	.0003	.0043	.0570	.0575	.0005	.0035	.0003	1.4864
.3051	.0760	.2465	.0054	.0066	.0005	.0071	.0760	.0767	.0006	.0045	.0004	1.4909
.3814	.0951	.2318	.0051	.0097	.0007	.0104	.0951	.0958	.0008	.0054	.0005	1.4944
.4576	.1141	.2183	.0049	.0133	.0009	.0142	.1141	.1150	.0009	.0063	.0006	1.4974
.5339	.1331	.2052	.0047	.0173	.0013	.0185	.1331	.1341	.0010	.0071	.0007	1.5003
.6102	.1521	.1922	.0046	.0216	.0016	.0232	.1521	.1532	.0011	.0079	.0007	1.5030
.6864	.1711	.1788	.0045	.0261	.0020	.0281	.1711	.1724	.0012	.0086	.0008	1.5056
.7627	.1901	.1644	.0044	.0308	.0024	.0332	.1901	.1915	.0013	.0093	.0009	1.5082
.8390	.2091	.1484	.0043	.0356	.0029	.0385	.2091	.2106	.0014	.0100	.0009	1.5108
.9153	.2281	.1282	.0043	.0401	.0034	.0436	.2281	.2297	.0015	.0107	.0010	1.5131
.9915	.2471	.1023	.0042	.0443	.0040	.0483	.2471	.2488	.0016	.0113	.0010	1.5158
1.0678	.2659	.0722	.0042	.0476	.0046	.0522	.2659	.2675	.0016	.0118	.0011	1.5197
1.1441	.2839	.0472	.0041	.0500	.0053	.0552	.2839	.2856	.0017	.0124	.0011	1.5262
1.2203	.3011	.0259	.0041	.0514	.0060	.0574	.3011	.3029	.0018	.0130	.0012	1.5332
1.2966	.3176	.0066	.0040	.0521	.0067	.0588	.3176	.3195	.0019	.0136	.0012	1.5398
1.3729	.3333	-.0111	.0039	.0520	.0075	.0595	.3333	.3354	.0020	.0143	.0013	1.5460
1.4492	.3484	-.0274	.0039	.0512	.0084	.0595	.3484	.3505	.0021	.0150	.0014	1.5517
1.5254	.3627	-.0427	.0038	.0498	.0092	.0590	.3627	.3649	.0022	.0157	.0014	1.5571
1.6017	.3762	-.0569	.0038	.0478	.0101	.0579	.3762	.3786	.0023	.0164	.0015	1.5620
1.6780	.3891	-.0701	.0037	.0452	.0110	.0563	.3891	.3916	.0024	.0171	.0016	1.5667
1.7542	.4012	-.0823	.0037	.0423	.0119	.0543	.4012	.4038	.0025	.0178	.0016	1.5711
1.8305	.4127	-.0935	.0036	.0391	.0129	.0520	.4127	.4153	.0027	.0185	.0017	1.5753
1.9068	.4234	-.1038	.0036	.0355	.0139	.0494	.4234	.4262	.0028	.0193	.0018	1.5792
1.9831	.4334	-.1130	.0035	.0318	.0149	.0467	.4334	.4363	.0029	.0201	.0018	1.5829
2.0593	.4427	-.1214	.0035	.0280	.0159	.0439	.4427	.4457	.0030	.0208	.0019	1.5864
2.1356	.4512	-.1287	.0034	.0241	.0169	.0410	.4512	.4544	.0032	.0217	.0020	1.5897
2.2119	.4591	-.1351	.0034	.0204	.0179	.0383	.4591	.4624	.0033	.0225	.0021	1.5928
2.2881	.4663	-.1403	.0033	.0167	.0190	.0357	.4663	.4698	.0034	.0233	.0021	1.5957
2.3644	.4728	-.1445	.0033	.0132	.0200	.0332	.4728	.4764	.0036	.0242	.0022	1.5984
2.4407	.4786	-.1475	.0033	.0100	.0211	.0311	.4786	.4823	.0037	.0251	.0023	1.6009
2.5169	.4837	-.1492	.0032	.0071	.0221	.0292	.4837	.4876	.0039	.0260	.0024	1.6032
2.5932	.4881	-.1496	.0032	.0045	.0232	.0277	.4881	.4921	.0040	.0270	.0025	1.6054

(c) Page 3.

Figure 14.- Continued.

TEST CASE L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE NPR=5.03 ITERATION NO 15

MO = .8000 TT = 324.44 KELVIN PT = 100720.0 PASCALS L = .152400 METERS SREF = .018242 SQ METERS

BOUNDARY LAYER SEPARATION AT X/L = 9.529286 BOUNDARY LAYER REATTACHMENT AT X/L = 11.633000

X/L	R/L	CP	CF	CDP	CDF	CDT	RDS/L	RC/L	DEL*/L	DEL/L	THETA/L	H
2.6695	.4918	-.1484	.0031	.0024	.0242	.0266	.4918	.4960	.0042	.0280	.0026	1.6075
2.7458	.4948	-.1452	.0031	.0006	.0252	.0258	.4948	.4992	.0044	.0290	.0027	1.6095
2.8220	.4972	-.1397	.0030	-.0007	.0262	.0255	.4972	.5018	.0046	.0301	.0029	1.6116
2.8983	.4988	-.1314	.0030	-.0016	.0272	.0256	.4988	.5036	.0048	.0313	.0030	1.6138
2.9746	.4997	-.1173	.0029	-.0021	.0282	.0261	.4997	.5048	.0051	.0325	.0032	1.6169
3.0508	.5000	-.0979	.0028	-.0022	.0291	.0270	.5000	.5054	.0054	.0339	.0033	1.6209
3.1271	.5000	-.0790	.0028	-.0022	.0300	.0279	.5000	.5058	.0058	.0353	.0035	1.6240
3.2034	.5000	-.0665	.0027	-.0022	.0309	.0288	.5000	.5060	.0060	.0366	.0037	1.6244
3.2797	.5000	-.0576	.0027	-.0022	.0318	.0296	.5000	.5063	.0063	.0380	.0039	1.6234
3.3559	.5000	-.0508	.0027	-.0022	.0326	.0305	.5000	.5065	.0065	.0393	.0040	1.6217
3.4322	.5000	-.0454	.0026	-.0022	.0335	.0313	.5000	.5067	.0067	.0406	.0041	1.6196
3.5085	.5000	-.0411	.0026	-.0022	.0343	.0321	.5000	.5069	.0069	.0419	.0043	1.6174
3.5847	.5000	-.0374	.0026	-.0022	.0351	.0329	.5000	.5071	.0071	.0432	.0044	1.6152
3.6610	.5000	-.0343	.0026	-.0022	.0359	.0338	.5000	.5073	.0073	.0444	.0045	1.6130
3.7373	.5000	-.0317	.0026	-.0022	.0367	.0346	.5000	.5074	.0074	.0457	.0046	1.6108
3.8136	.5000	-.0294	.0026	-.0022	.0375	.0354	.5000	.5076	.0076	.0469	.0047	1.6087
3.8898	.5000	-.0275	.0026	-.0022	.0383	.0362	.5000	.5078	.0078	.0481	.0048	1.6068
3.9661	.5000	-.0257	.0025	-.0022	.0391	.0370	.5000	.5079	.0079	.0493	.0050	1.6049
4.0424	.5000	-.0242	.0025	-.0022	.0399	.0377	.5000	.5081	.0081	.0504	.0051	1.6031
4.1186	.5000	-.0229	.0025	-.0022	.0407	.0385	.5000	.5083	.0083	.0516	.0052	1.6015
4.1949	.5000	-.0217	.0025	-.0022	.0415	.0393	.5000	.5084	.0084	.0528	.0053	1.5999
4.2712	.5000	-.0206	.0025	-.0022	.0423	.0401	.5000	.5086	.0086	.0539	.0054	1.5985
4.3475	.5000	-.0197	.0025	-.0022	.0430	.0409	.5000	.5088	.0088	.0550	.0055	1.5971
4.4237	.5000	-.0188	.0025	-.0022	.0438	.0416	.5000	.5089	.0089	.0562	.0056	1.5958
4.5000	.5000	-.0180	.0025	-.0022	.0446	.0424	.5000	.5091	.0091	.0573	.0057	1.5946
4.5763	.5000	-.0173	.0025	-.0022	.0453	.0432	.5000	.5092	.0092	.0584	.0058	1.5935
4.6525	.5000	-.0167	.0025	-.0022	.0461	.0439	.5000	.5094	.0094	.0595	.0059	1.5924
4.7288	.5000	-.0161	.0025	-.0022	.0469	.0447	.5000	.5095	.0095	.0605	.0060	1.5914
4.8051	.5000	-.0156	.0025	-.0022	.0476	.0455	.5000	.5097	.0097	.0616	.0061	1.5905
4.8814	.5000	-.0152	.0024	-.0022	.0484	.0462	.5000	.5098	.0098	.0627	.0062	1.5896
4.9576	.5000	-.0148	.0024	-.0022	.0491	.0470	.5000	.5100	.0100	.0638	.0063	1.5888
5.0339	.5000	-.0144	.0024	-.0022	.0499	.0477	.5000	.5101	.0101	.0648	.0064	1.5880
5.1102	.5000	-.0141	.0024	-.0022	.0506	.0485	.5000	.5103	.0103	.0658	.0065	1.5872
5.1864	.5000	-.0138	.0024	-.0022	.0514	.0492	.5000	.5104	.0104	.0669	.0066	1.5865
5.2627	.5000	-.0135	.0024	-.0022	.0521	.0500	.5000	.5106	.0106	.0679	.0067	1.5859

(d) Page 4.

Figure 14.- Continued.

TEST CASE L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE NPR=5.03 ITERATION NO 15

M0 = .8000 TT = 324.44 KELVIN PT = 100720.0 PASCALS L = .152400 METERS SREF = .018242 SQ METERS

BOUNDARY LAYER SEPARATION AT X/L = 9.529286 BOUNDARY LAYER REATTACHMENT AT X/L = 11.633000

X/L	R/L	CP	CF	CDP	CDF	CDT	RDS/L	RC/L	DEL*/L	DEL/L	THETA/L	H
5.3390	.5000	-.0133	.0024	-.0022	.0529	.0507	.5000	.5107	.0107	.0689	.0067	1.5852
5.4153	.5000	-.0131	.0024	-.0022	.0536	.0514	.5000	.5108	.0108	.0700	.0068	1.5846
5.4915	.5000	-.0130	.0024	-.0022	.0543	.0522	.5000	.5110	.0110	.0710	.0069	1.5841
5.5678	.5000	-.0128	.0024	-.0022	.0551	.0529	.5000	.5111	.0111	.0720	.0070	1.5835
5.6441	.5000	-.0127	.0024	-.0022	.0558	.0536	.5000	.5113	.0113	.0730	.0071	1.5830
5.7203	.5000	-.0127	.0024	-.0022	.0565	.0544	.5000	.5114	.0114	.0740	.0072	1.5825
5.7966	.5000	-.0126	.0024	-.0022	.0573	.0551	.5000	.5116	.0116	.0750	.0073	1.5820
5.8729	.5000	-.0126	.0024	-.0022	.0580	.0558	.5000	.5117	.0117	.0760	.0074	1.5816
5.9492	.5000	-.0126	.0023	-.0022	.0587	.0566	.5000	.5118	.0118	.0769	.0075	1.5812
6.0254	.5000	-.0127	.0023	-.0022	.0594	.0573	.5000	.5120	.0120	.0779	.0076	1.5807
6.1017	.5000	-.0127	.0023	-.0022	.0602	.0580	.5000	.5121	.0121	.0789	.0077	1.5803
6.1780	.5000	-.0128	.0023	-.0022	.0609	.0587	.5000	.5122	.0122	.0799	.0077	1.5800
6.2542	.5000	-.0130	.0023	-.0022	.0616	.0594	.5000	.5124	.0124	.0808	.0078	1.5796
6.3305	.5000	-.0131	.0023	-.0022	.0623	.0602	.5000	.5125	.0125	.0818	.0079	1.5793
6.4068	.5000	-.0133	.0023	-.0022	.0630	.0609	.5000	.5126	.0126	.0827	.0080	1.5789
6.4831	.5000	-.0135	.0023	-.0022	.0637	.0616	.5000	.5128	.0128	.0837	.0081	1.5786
6.5593	.5000	-.0138	.0023	-.0022	.0645	.0623	.5000	.5129	.0129	.0846	.0082	1.5783
6.6356	.5000	-.0141	.0023	-.0022	.0652	.0630	.5000	.5130	.0130	.0855	.0083	1.5779
6.7119	.5000	-.0144	.0023	-.0022	.0659	.0637	.5000	.5132	.0132	.0865	.0083	1.5776
6.7881	.5000	-.0148	.0023	-.0022	.0666	.0644	.5000	.5133	.0133	.0874	.0084	1.5773
6.8644	.5000	-.0153	.0023	-.0022	.0673	.0651	.5000	.5134	.0134	.0883	.0085	1.5770
6.9407	.5000	-.0157	.0023	-.0022	.0680	.0658	.5000	.5135	.0135	.0892	.0086	1.5767
7.0169	.5000	-.0163	.0023	-.0022	.0687	.0665	.5000	.5137	.0137	.0901	.0087	1.5764
7.0932	.5000	-.0169	.0023	-.0022	.0694	.0672	.5000	.5138	.0138	.0911	.0087	1.5762
7.1695	.5000	-.0176	.0023	-.0022	.0701	.0679	.5000	.5139	.0139	.0920	.0088	1.5759
7.2458	.5000	-.0184	.0023	-.0022	.0708	.0686	.5000	.5140	.0140	.0928	.0089	1.5756
7.3220	.5000	-.0192	.0023	-.0022	.0715	.0693	.5000	.5141	.0141	.0937	.0090	1.5753
7.3983	.5000	-.0202	.0023	-.0022	.0722	.0700	.5000	.5143	.0143	.0946	.0091	1.5749
7.4746	.5000	-.0213	.0023	-.0022	.0729	.0707	.5000	.5144	.0144	.0955	.0091	1.5746
7.5508	.5000	-.0225	.0023	-.0022	.0736	.0714	.5000	.5145	.0145	.0963	.0092	1.5743
7.6271	.5000	-.0239	.0022	-.0022	.0743	.0721	.5000	.5146	.0146	.0972	.0093	1.5739
7.7034	.5000	-.0255	.0022	-.0022	.0750	.0728	.5000	.5147	.0147	.0981	.0093	1.5735
7.7797	.5000	-.0273	.0022	-.0022	.0757	.0735	.5000	.5148	.0148	.0989	.0094	1.5731
7.8559	.5000	-.0294	.0022	-.0022	.0764	.0742	.5000	.5149	.0149	.0997	.0094	1.5727
7.9322	.5000	-.0317	.0022	-.0022	.0771	.0749	.5000	.5149	.0149	.1005	.0095	1.5722

(e) Page 5.

Figure 14.- Continued.

TEST CASE L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE NPR=5.03 ITERATION NO 15

MO = .8000 TT = 324.44 KELVIN PT = 100720.0 PASCALS L = .152400 METERS SREF = .018242 SQ METERS

BOUNDARY LAYER SEPARATION AT X/L = 9.529286 BOUNDARY LAYER REATTACHMENT AT X/L = 11.633000

X/L	R/L	CP	CF	CDP	CDF	CDT	RDS/L	RC/L	DEL*/L	DEL/L	THETA/L	H
8.0085	.5000	-.0344	.0022	-.0022	.0778	.0756	.5000	.5150	.0150	.1013	.0095	1.5716
8.0847	.5000	-.0376	.0022	-.0022	.0785	.0763	.5000	.5151	.0151	.1021	.0096	1.5710
8.1610	.5000	-.0413	.0022	-.0022	.0792	.0770	.5000	.5151	.0151	.1029	.0096	1.5703
8.2373	.5000	-.0457	.0023	-.0022	.0799	.0777	.5000	.5151	.0151	.1036	.0097	1.5695
8.3136	.5000	-.0512	.0023	-.0022	.0806	.0784	.5000	.5152	.0152	.1043	.0097	1.5686
8.3898	.5000	-.0570	.0023	-.0022	.0813	.0792	.5000	.5151	.0152	.1050	.0097	1.5677
8.4661	.5000	-.0652	.0023	-.0022	.0820	.0799	.5000	.5151	.0151	.1056	.0096	1.5663
8.5424	.5000	-.0745	.0023	-.0022	.0828	.0806	.5000	.5150	.0150	.1062	.0096	1.5650
8.6186	.5000	-.0868	.0023	-.0022	.0835	.0813	.5000	.5149	.0149	.1067	.0095	1.5633
8.6949	.5000	-.1029	.0023	-.0022	.0843	.0821	.5000	.5147	.0147	.1071	.0094	1.5613
8.7712	.5000	-.1246	.0024	-.0022	.0850	.0829	.5000	.5144	.0144	.1074	.0092	1.5590
8.8475	.5000	-.1538	.0024	-.0022	.0858	.0837	.5000	.5139	.0140	.1076	.0090	1.5566
8.9237	.5000	-.2017	.0025	-.0022	.0867	.0845	.5000	.5132	.0133	.1074	.0086	1.5543
9.0000	.5000	-.2803	.0026	-.0022	.0876	.0854	.5000	.5120	.0123	.1069	.0079	1.5546
9.0500	.4991	-.3242	.0027	-.0011	.0882	.0871	.4991	.5103	.0118	.1071	.0076	1.5587
9.1000	.4965	-.3423	.0027	.0024	.0888	.0912	.4965	.5075	.0118	.1078	.0075	1.5623
9.1500	.4921	-.3434	.0027	.0083	.0895	.0978	.4921	.5033	.0120	.1090	.0077	1.5647
9.2000	.4859	-.3422	.0026	.0166	.0901	.1067	.4859	.4978	.0123	.1106	.0078	1.5668
9.2500	.4780	-.3230	.0026	.0268	.0907	.1175	.4780	.4908	.0129	.1129	.0082	1.5682
9.3000	.4681	-.2700	.0025	.0379	.0913	.1291	.4681	.4823	.0141	.1162	.0090	1.5700
9.3500	.4565	-.1939	.0023	.0479	.0918	.1397	.4565	.4725	.0160	.1206	.0102	1.5763
9.4000	.4429	-.1081	.0021	.0553	.0922	.1475	.4429	.4617	.0187	.1266	.0118	1.5915
9.4500	.4273	-.0254	.0019	.0590	.0926	.1516	.4273	.4502	.0222	.1342	.0137	1.6177
9.5000	.4096	.0471	.0017	.0584	.0929	.1513	.4096	.4387	.0265	.1435	.0160	1.6543
9.5500	.3899	.1034	.0015	.0537	.0931	.1468	.3930	.4275	.0311	.1533	.0184	1.6952
9.6000	.3679	.1426	.0014	.0455	.0933	.1388	.3806	.4170	.0352	.1617	.0203	1.7320
9.6500	.3436	.1621	.0013	.0350	.0935	.1285	.3681	.4073	.0382	.1694	.0218	1.7521
9.7000	.3168	.1632	.0013	.0235	.0937	.1171	.3559	.3986	.0396	.1761	.0226	1.7496
9.7500	.2873	.1619	.0013	.0119	.0938	.1057	.3439	.3914	.0408	.1829	.0234	1.7444
9.8000	.2550	.1518	.0013	.0009	.0939	.0948	.3319	.3851	.0412	.1893	.0238	1.7288
9.8500	.2867	.1418	.0014	-.0006	.0940	.0935	.3520	.3790	.0379	.1784	.0221	1.7147
9.9000	.2871	.1391	.0014	-.0006	.0942	.0936	.3399	.3730	.0390	.1853	.0228	1.7093
9.9500	.2869	.1395	.0014	-.0006	.0943	.0937	.3272	.3666	.0406	.1933	.0238	1.7074
10.0000	.2861	.1538	.0013	-.0006	.0944	.0938	.3136	.3599	.0438	.2038	.0255	1.7208
10.0500	.2849	.1899	.0012	-.0006	.0945	.0939	.2996	.3531	.0503	.2180	.0285	1.7645

attached

separated

$C_f > 0$ in separated region

(f) Page 6.

Figure 14.- Continued.

TEST CASE L/D=9 FOREBODY L/D=0,8 DB/D=0,51 CIRCULAR ARC NOZZLE NPR=5,03 ITERATION NO 15

MO = ,8000 TT = 324,44 KELVIN PT = 100720,0 PASCALS L = ,152400 METERS SREF = ,018242 SQ METERS

BOUNDARY LAYER SEPARATION AT X/L = 9,529286 BOUNDARY LAYER REATTACHMENT AT X/L = 11,633000

Separated

X/L	R/L	CP	CF	CDP	CDF	CDT	RDS/L	RC/L	DEL*/L	DEL/L	THETA/L	H
10,1000	,2840	,2065	,0011	-,0006	,0946	,0940	,2858	,3469	,0552	,2312	,0309	1,7863
10,1500	,2839	,2083	,0011	-,0006	,0947	,0941	,2835	,3422	,0558	,2340	,0313	1,7858
10,2000	,2843	,2037	,0011	-,0006	,0948	,0942	,2839	,3389	,0550	,2338	,0310	1,7753
10,2500	,2850	,1863	,0012	-,0006	,0949	,0943	,2845	,3366	,0523	,2317	,0299	1,7471
10,3000	,2857	,1634	,0013	-,0006	,0950	,0944	,2853	,3349	,0492	,2292	,0286	1,7163
10,3500	,2862	,1458	,0013	-,0006	,0952	,0946	,2859	,3334	,0470	,2276	,0277	1,6957
10,4000	,2866	,1322	,0014	-,0006	,0953	,0947	,2864	,3322	,0455	,2266	,0271	1,6811
10,4500	,2869	,1201	,0014	-,0006	,0954	,0948	,2868	,3312	,0443	,2259	,0265	1,6691
10,5000	,2872	,1085	,0015	-,0006	,0955	,0950	,2872	,3303	,0431	,2252	,0260	1,6584
10,5500	,2875	,0981	,0015	-,0006	,0957	,0951	,2875	,3297	,0422	,2248	,0256	1,6496
10,6000	,2878	,0888	,0015	-,0006	,0958	,0952	,2877	,3292	,0414	,2244	,0252	1,6420
10,6646	,2881	,0785	,0015	-,0006	,0960	,0954	,2881	,3286	,0405	,2241	,0248	1,6342
10,7291	,2884	,0696	,0016	-,0006	,0962	,0956	,2883	,3282	,0398	,2240	,0245	1,6277
10,7937	,2886	,0621	,0016	-,0006	,0964	,0958	,2886	,3279	,0393	,2241	,0242	1,6225
10,8583	,2888	,0555	,0016	-,0006	,0966	,0960	,2888	,3276	,0388	,2242	,0240	1,6180
10,9228	,2889	,0500	,0016	-,0006	,0968	,0962	,2889	,3274	,0385	,2244	,0238	1,6144
10,9874	,2891	,0447	,0016	-,0006	,0970	,0964	,2891	,3272	,0381	,2246	,0237	1,6111
11,0519	,2892	,0407	,0016	-,0006	,0972	,0966	,2892	,3271	,0379	,2250	,0236	1,6085
11,1165	,2893	,0368	,0016	-,0006	,0974	,0968	,2893	,3270	,0377	,2253	,0235	1,6061
11,1811	,2894	,0335	,0017	-,0006	,0976	,0970	,2894	,3269	,0375	,2257	,0234	1,6040
11,2456	,2895	,0306	,0017	-,0006	,0978	,0972	,2895	,3269	,0374	,2262	,0233	1,6022
11,3102	,2896	,0277	,0017	-,0006	,0980	,0974	,2896	,3269	,0373	,2266	,0233	1,6005
11,3748	,2897	,0254	,0017	-,0006	,0982	,0976	,2896	,3268	,0372	,2271	,0233	1,5990
11,4393	,2897	,0232	,0017	-,0006	,0985	,0979	,2897	,3268	,0371	,2276	,0232	1,5977
11,5039	,2898	,0211	,0017	-,0006	,0987	,0981	,2898	,3268	,0370	,2281	,0232	1,5964
11,5684	,2898	,0191	,0017	-,0006	,0989	,0983	,2898	,3268	,0370	,2286	,0232	1,5952
11,6330	,2899	,0171	,0017	-,0006	,0991	,0985	,2899	,3268	,0369	,2290	,0232	1,5940

FOR ITERA= 16 8 ITERATIONS REQUIRED FOR CONVERGENCE IN MISNA2

(g) Page 7.

Figure 14.- Continued.

POTENTIAL FLOW SOLUTION

TEST CASE L/D=9 FOREBODY L/D=0.8 DB/D=0.51 CIRCULAR ARC NOZZLE NPR=5.03

OFF-BODY UNIFORM AXISYMMETRIC FLOW

	X/L	R/L	VX	VR	VT	ETA	ML	CP
1	8.800000	.600000	1.062181	-.009934	1.062227	-.535866	.856848	-.125713
2	9.000000	.600000	1.111166	-.035165	1.111723	-1.812641	.903119	-.227155
3	9.200000	.600000	1.119641	-.109287	1.124962	-5.574925	.915665	-.254448
4	9.400000	.600000	1.029521	-.147716	1.040064	-8.165084	.836438	-.080670
5	9.600000	.600000	.956023	-.118833	.963380	-7.085490	.767182	.072730
6	9.800000	.600000	.943930	-.082651	.947542	-5.004080	.753125	.103845
7	9.800000	.550000	.937734	-.089026	.941950	-5.423231	.748182	.114777
8	9.800000	.500000	.931133	-.095801	.936049	-5.874312	.742975	.126285
9	9.800000	.450000	.924368	-.102942	.930082	-6.354575	.737722	.137886
10	9.800000	.400000	.917520	-.110636	.924166	-6.875620	.732524	.149355

(h) Page 8.

Figure 14.- Concluded.

1 Report No	2 Government Accession No	3 Recipient's Catalog No
NASA TM-78779		

4 Title and Subtitle	5 Report Date
DONBOL: A COMPUTER PROGRAM FOR PREDICTING AXISYMMETRIC NOZZLE AFTERBODY PRESSURE DISTRIBUTIONS AND DRAG AT SUBSONIC SPEEDS	May 1979
	6 Performing Organization Code
7 Author(s)	8 Performing Organization Report No
Lawrence E. Putnam	L-12658
	10 Work Unit No
9 Performing Organization Name and Address	505-04-13-01
NASA Langley Research Center Hampton, VA 23665	11 Contract or Grant No
	13 Type of Report and Period Covered
12 Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, DC 20546	14 Sponsoring Agency Code

15 Supplementary Notes

16 Abstract

A Neumann solution for inviscid external flow has been coupled to a modified Reshotko-Tucker integral boundary-layer technique, the control volume method of Presz for calculating flow in the separated region, and an inviscid one-dimensional solution for the jet exhaust flow in order to predict axisymmetric nozzle afterbody pressure distributions and drag. The viscous and inviscid flows are solved iteratively until convergence is obtained. A computer algorithm of this procedure has been written and is called DONBOL. This paper provides a description of the computer program and a guide to its use. Comparisons of the predictions of this method with experiment show that the method accurately predicts the pressure distributions of boattail afterbodies which have the jet exhaust flow simulated by solid bodies. For nozzle configurations which have the jet exhaust simulated by high-pressure air, the present method significantly underpredicts the magnitude of nozzle pressure drag. This deficiency results because the method neglects the effects of jet plume entrainment. This method is limited to subsonic free-stream Mach numbers below that for which the flow over the body of revolution becomes sonic.

17 Key Words (Suggested by Author(s))	18 Distribution Statement
Nozzle drag Pressure drag Body of revolution Jet exhaust flow	FEDD Distribution Subject Category 02

19 Security Classif (of this report)	20 Security Classif (of this page)	21 No of Pages	22 Price
Unclassified	Unclassified	97	

Available: NASA's Industrial Application Centers